PLATE-Seq: An Efficient and Scalable Method for Using RNA-Seq as a Primary Output in High Throughput Drug Screens

Forest Ray

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2016

Table of Contents

List of Figures

Acknowledgements

Although only a single author is listed on a thesis dissertation, success in an undertaking so psychologically crippling comes from the combined efforts of many people. I am both proud and fortunate to have been surrounded by a strong and supportive group of friends, family and co-workers throughout my graduate trials. First and foremost, I must thank my wife, Jordan. She, above all others, has pushed me to continue, where I otherwise surely would have walked away in defeat. I can only hope that I might one day repay her in kind. Without the help and guidance of Alexander Lachmann and Federico Giorgi, I simply would not have been able to transition from the wet lab to the computational side. I may never meet another mind quite like that of Mariano Alvarez. I have worked alongside him for six long years and although I still cannot quite follow his thought process, I am indebted to him for all the help that he has rendered along the way. My thesis committee has been instrumental in keeping me on track. Peter Sims has provided invaluable aid and mentorship over the course of this spiritual march towards Mordor. It is with some concern that I consider the possibility of never again having a mentor of his caliber. Saeed Tavazoie's clear guidance and insights have been invaluable in moving my project forward. I am further indebted to my thesis advisor, Andrea Califano, who gave me wider latitude to experiment and fail than I am ever likely to find again. It wasn't pretty, but by God, I survived.

Chapter 1

Background and Introduction

The identification of drug treatments that are useful in diverse therapeutic settings is a significant driving force in biomedical research [55], [68], [48]. Typical means for measuring the efficacy of a drug for a given clinical application include protein-protein interactions, cell death, mitochondrial respiration and cell growth as well as broader measurements of absorption, distribution, metabolism, excretion and toxicity (ADMET), specifically related the the drug or drugs being tested [85]. A wide array of methods are routinely employed to perform these screens, from ligand binding assays [93] to high-throughput proteomics [91]. One method that is currently underutilized in small-molecule drug screens and drug discovery is high-throughput transcriptome sequencing, such as RNA-Seq. Although RNA-Seq is routinely used to profile patterns of genetic changes following perturbations such as drug treatment [99], it has not, to my knowledge, yet been used as the primary readout of a drug screen.

RNA-seq has been used in addition to other biochemical assays as part of an integrated screening pipeline, as in [33], but not as the primary, stand-alone readout of the screen. In contrast to the previously mentioned drug screen measurements, RNA-Seq is a much more scalable technology. The primary impediment to making full use of this potential scalability, however, is the price of sequencing, which, while continuing to decline, remains prohibitively expensive for many applications. Performing large-scale drug screens is one highly desirable application.

Ultimately, a drug screen encompasses multiple perturbations that have genome-

wide effects on cellular networks. The lack of a genome-wide readout is, therefore, a shortcoming in that it represents a lack of potentially very valuable information. This lack is understandable, however, as the cost of sequencing large screens, which must frequently take into account multiple drug concentrations across several time points, is considerable. As we progress in our ability to interrogate cellular networks at a systems level, however, vital techniques such as large-scale drug screens stand to benefit from advances in network biology. To do so, however, they will require genome-scale outputs.

One way to drive down the cost of multiplexing greater numbers of treatments into a drug screen is to lower the depth at which an expression library is sequenced, thereby allowing more samples to be sequenced in any given run. There has been considerable interest in making increasingly multiplexed expression libraries to enable researchers to perform increasingly complex and larger-scale sequencing experiments, particularly in the field of single-cell RNA-Seq, where samples must necessarily be barcoded and pooled [37], [40], [38]. A question that is central to this investigation is that of how little sequencing depth is needed to reliably measure gene expression signatures. In this dissertation, I will describe a method for transcriptome profiling at the relatively low depth of 500,000 to 2 million reads per sample that enables increased sample multiplicity and is amenable to network biology methods. Because this methodology rests on the availability of appropriate gene regulatory networks with which to interpret transcriptome library sequencing results, I will devote some time to describing what those networks are and how they are inferred.

1.1 The Biological Importance of Gene Regulatory Networks

A gene regulatory network (GRN) is a conceptual framework for modelling and characterizing complex interactions between diverse components of a biological system. [100]. Gene regulatory networks are critically important to the field of systems biology, as they serve as a roadmap for understanding developmental and regulatory biological processes [102] and for predicting cellular responses to external stimuli [24]. Differential gene expression forms the basis of studies that seek to reverse engineer GRNs. Global patterns of differentially expressed genes form a higher-order structure called gene expression

profiles (GEPs). Because different phenotypic states are associated with characteristic gene expression profiles, these profiles can be used to characterize specific phenotypic states, such as responses to experimental perturbations. In this way, one can characterize the effect of a give perturbation on cellular phenotypes and compare different perturbations based on the impact that each has on GEPs. Given a specific target GEP, one describing a drug susceptible state, for instance, drugs or other small molecules can even be screened for their ability to induce that profile.

1.2 Inferring Gene Regulatory Networks

In any differential gene expression dataset, statistical associations such as coexpression or co-repression between various genes are sought to identify those genes that may have relevant regulatory interactions. Although coexpression cannot unambiguously disentangle cause from effect, nor differentiate direct from indirect interactions, either type of interaction contains potentially useful information. Sets of genes that are differentially expressed or repressed following a given perturbation may form functional modules that describe key biological processes, such as cell cycle arrest or apoptosis, either of which are important therapeutic target outcomes for a number of pathological conditions [21], [27], [82]. Gene expression profiles can also be integrated with data from other sources, from tandem mass spectrometry (MS-MS) and protein-protein interaction (PPI) inference algorithms to help identify candidate physical interactions that can be validated biochemically.

1.2.1 Coexpression

Coexpression is arguably the simplest way to screen for potential genetic interactions. Although many methods for generating coexpression-based networks have been proposed, they all essentially follow the same two-step approach. Coexpression is measured in the first step, with each pair of genes being assigned a similarity score, with the exact nature of this score varying by method. The second step consists of setting a significance threshold and connecting all genes with similarity values higher than the threshold by edges in the resulting coexpression graph, an example of which can be seen in 1.1, from [94]. Common

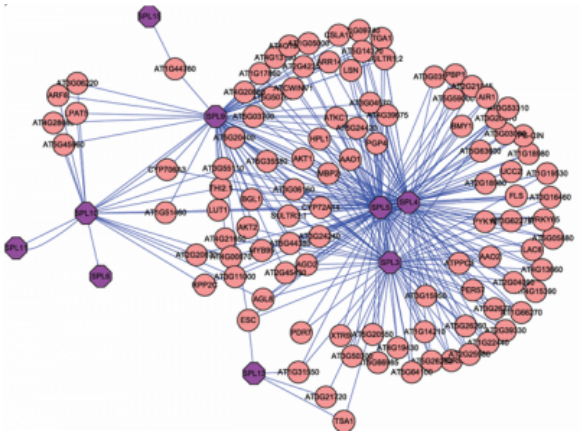

Figure 1.1: A coexpression network built around the SBP-box Gene in Arabidopsis. Wang, et al., 2009.

measures for scoring gene expression similarity include the Pearson correlation coefficient, Spearman rank correlation, euclidean distance, partial correlation [92], regression [67] and more recently, mutual information. Each method for measuring coexpression brings its own strengths and weaknesses to the analysis. The Pearson correlation coefficient is one of the most widely-used means of estimating covariance in gene expression matrices. It takes on values ranging from -1 to 1, where anything with an absolute value near one signifies a strong correlation. Two drawbacks to the Pearson correlation coefficient are that it can be applied to only linear relationships and that it is sensitive to outliers. Gene regulatory relationships are rarely linear [34] and expression data is noisy and rife with outlying values. In contrast to the Pearson coefficient, the Spearman rank correlation is computed on ranked values and describes monotonic, rather than linear relationships. Euclidean distance, while simple to implement, risks a high false positive rate from assigning significance to genes, whose expression is consistently low, but are otherwise only weakly correlated or uncorrelated.

1.2.2 Bayesian Networks

A Bayesian network (BN) is another method for inferring and visualizing gene regulatory relationships. As a directed acyclic graph (DAG), Bayesian networks express the direction of each relationship (A operates on B, or vice versa) and does not allow loops

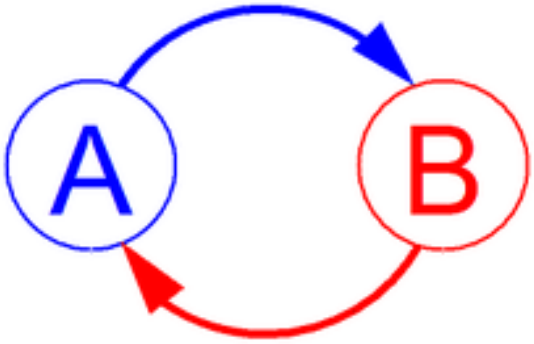

Figure 1.2: A generic representation of a feedback loop, in which A activates B, which in turn activates A. Source: Wikipedia.

in the network, which allows for efficient inference algorithms [63]. The expression of each gene in a BN is represented as a conditional probability, depending on other genes in the dataset. Formally, for a pair of genes A and B,

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \tag{1.1}$$

where P(A) and P(B) are the marginal probabilities of A and B, P(A | B) is the conditional probability, or that of observing A given that B is true and finally, P(B | A) is the probability of observing B given that A is true.

Where biological systems do contain feedback loops, acyclic graphs do not serve as their best representation. One means for overcoming this limitation has been to incorporate time series data into Bayesian inference algorithms. The resulting networks are called dynamic Bayesian networks and are better able to recapitulate biological network features, such as feedback loops (Fig. 1.2) [7]. One drawback to using them, however, is that they are not simple to implement, as they require many input parameters, which often require machine learning methods to acquire. Although the requirement for many parameters can make dynamic Bayesian networks difficult and computationally expensive to implement for complete networks, they frequently find use in modelling sub-networks.

1.2.3 Mutual Information Networks

More recently, information theoretic approaches have been applied to inferring biological networks. Mutual information (MI) has emerged as a robust way to measure indirect, non-linear and non-monotonic relationships. Formally, the mutual information between two discrete variables X and Y is defined as

$$I(X;Y) = \sum_{y}\sum_{x} p(x,y) log \frac{p(x,y)}{p(x)p(y)}, \tag{1.2}$$

where p(x,y) is the probability of both X and Y being in the same state (their joint probability), and p(x) and p(y) are the probabilities of either one being in said state (their marginal probabilities). Intuitively, this ratio represents how much the knowledge of the state of one of those variables reduces the uncertainty concerning the state of the other variable. At the two extremes, this means that if X and Y are completely independent, then knowing X provides no information concerning Y and vice versa, while if one is entirely deterministic of the other, then knowing one implies knowing the state of the other completely. By operating under the assumption of complete independence between two genes, mutual information then seeks to measure their inherent dependence. Formally, this means that I(X;Y) = 0 if and only if X and Y are completely independent of each other. This is easy to see in the case of complete independence, where p(x,y) = p(x)p(y). In that case,

$$log \frac{p(x,y)}{p(x)p(y)} = log 1 = 0. \tag{1.3}$$

One potential disadvantage of using mutual information is the reliance on a large number of input samples. The large sample number requirement for MI calculation occurs because of the need to estimate the joint probability distribution of the expression of all gene pairs. Additionally, the estimation of higher-order interactions can lead to spurious detection of sophisticated non-linear relationships that are not biologically meaningful. Despite these drawbacks, MI has proven to be a useful and robust method for inferring GRNs that has been meeting with success in the field [51], [4].

The Califano lab has developed an MI-based algorithm called the Algorithm for the Reconstruction of Accurate Networks (ARACNe) [56] to infer GRNs and in particular, the interactions between TFs and their transcriptional targets. Because gene expression is

regulated by TFs and their cofactors, which are themselves gene products, ARACNe rests on the assumption that statistical associations between gene mRNA abundances should be informative of, although not directly proportional to, protein activity. ARACNe works by estimating the MI for interactions between all TF-target pairs (edges in a graph), setting an MI threshold and applying the data processing inequality (DPI) to prune edges. The DPI assumes that for any two genes that interact only through a third gene, the smallest MI value for any of the three possible interactions must come from an indirect interaction, which can be left aside in the search for direct TF-target interactions. Targets of TFs are then identified as genes, that share high MI with the expression of a TF. Sets of targets that are coordinately associated with a given TF are assumed to belong to the regulatory program of that TF. These sets of TF-specific targets are known as regulons. This description is meant to serve more as a description of the theory underpinning ARACNe, than as a mechanistic explanation of how it works. ARACNe will be described in greater mechanistic detail in the methods section, in Chapter Two. Because the networks built by ARACNe provide details of which genes are targets of given TFs for specific cellular contexts, a natural extension of this research was to use target mRNA abundances as a means of reporting on TF activity. Where the targets of a certain TF are observed to be differentially expressed, one can assume that the TF governing the expression of those targets is differentially active. This is important because TFs are key drivers of cellular phenotypes and although alterations affecting them are implicated in a number of disorders [25], [43], these changes frequently occur post-translationally [26], [88], [45] and are therefore not directly captured in expression data. Transcript abundances of a TF-specific regulon can fill this knowledge gap, as they provide a direct link between target expression and TF activity.

A key feature of ARACNe-inferred networks is that they can be interrogated to find the TFs, whose regulon enrichment patterns are indicative of specific molecular phenotypes. The TFs responsible for the enriched regulons then theoretically describe the minimum set of TFs required to achieve that phenotype.

1.3 Applications of Gene Regulatory Networks

Simply being able to infer a GRN is not an end in and of itself. The utility of these networks lies in being able to facilitate solutions for diverse biological and biomedical questions.

1.3.1 Mapping the Causality of Genetic Interactions

The observation of how mRNA abundances change in response to experimental perturbations, or simply over the course of time and development, provide information on potential regulatory relationships between genes. As has been described in the sections concerning how networks are inferred, these shifting expression patterns can be used to determine which gene sets describe specific phenotypes or are used in certain biological pathways.

1.3.2 Gene Regulatory Networks as Diagnostic Tools

Recently, the idea of using GRNs as diagnostic or prognostic tools has gained increasing traction [15], [8], [11], [20]. This is a particularly attractive application for complex disorders like cancer, which are represented by sets of interacting genes and their related pathways, rather than by individual genes [35]. In this case, rather than relying on a small number of defined genes, the network, or more likely a clinically relevant sub-network, would serve as a biomarker for the condition. Ideally, network-based biomarkers would outperform their individual gene counterparts by being better able to account for the interaction structure between the genes in the network. The need to consider systems-level interaction changes is a feature not only of cancer, but of complex disorders in general, suggesting that we can expect to see GRNs increasingly applied as diagnostic and prognostic tools in these contexts as well. Nonetheless, developing accurate network-based biomarkers for complex disorders remains a challenging task. Of the numerous obstacles to achieving this goal, two of the most immediate are the need for large datasets covering multiple experimental conditions and the cost of obtaining these datasets. The Califano lab has lately been pursuing an interesting idea that may ease the acquisition of these datasets by lowering the amount

of per-sample sequencing needed for analysis. Because the transcripts that are sequenced in any RNA-Seq run are the transcriptional targets of active transcription factors, we reasoned that the activity of these factors could be inferred from the differnetial expression of their targets, an idea that will be discussed in greater detail in the following section.

1.3.3 Transcription Factors as Therapeutic Targets

Because of the role that transcription factors play in regulating cell responses, they make desirable therapeutic targets. However, transcription factors are considerably challenging to operate on directly for a number of reasons [28]. First, they bind DNA, which is negatively charged. This means that TFs tend to carry a positive charge, necessitating that any molecule designed to bind to them carry a negative charge. Since the cell membrane functions as a selective barrier to charged molecules, simply getting these theoretical TF-targeting molecules into the cell presents one challenge. Second, TFs do not typically bind small, distinct ligands. Kinases, for instance, bind ATP, so most of the drugs that target kinases inhibit them by binding to their ATP binding pocket better than ATP can. Metabolic enzymes typically bind small molecule metabolites, and can be interfered with by making drugs that look like their metabolite of interest. Without a well-defined small molecule interaction surface, drug makers are left to deal with the much greater interaction surfaces that TFs use to interact with DNA and with multiple co-factors. Targeting such a comparatively large interaction surface would necessitate designing similarly large drugs, which would be mechanically hindered from accessing the cell's interior. The fact that TFs generally act in complex with other proteins presents yet another challenge to targeting them directly. A final challenge in targeting TFs is in designing efficient, high-throughput screening assays for molecules that can target them. To look for kinase inhibitors, for example, one can design an assay that looks for ATP hydrolysis. Transcription factors lack any such straightforward and uniform enzymatic activity, necessitating more complex, expensive and lower-throughput assays. RNA interference-based strategies have been proposed to target TF expression in disease contexts, but effectively delivering these therapies in a clinical setting remains problematic [98], [42], [19]. A screen that allows us to control them based on how drugs alter their activity may be the most actionable strategy possible.

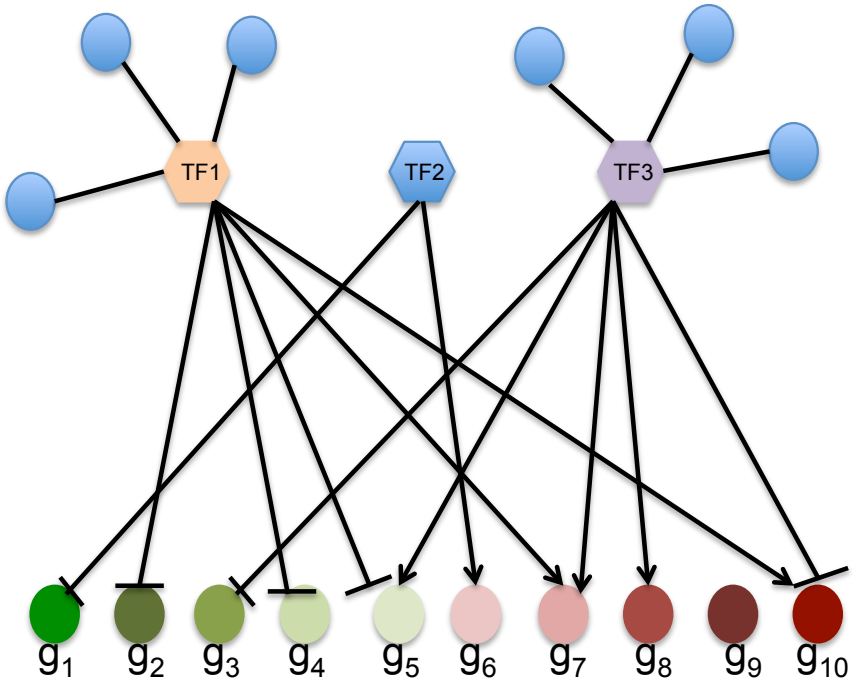

Figure 1.3: Transcription factor pleiotropy: each TF regulates the expression of multiple downstream targets (g_i).

To this end, the Califano lab has pioneered the Virtual Inference of Protein-activity by Enriched Regulon analysis (VIPER) algorithm, which estimates TF activity from regulon transcript abundances given a cell context-specific ARACNe-inferred network (Alvarez, et al., under review).

1.4 Inferring Transcription Factor Activity From Gene Regulatory Networks

Transcription factors operate pleiotropically, in that each TF regulates the expression of multiple transcriptional targets, as shown in Fig. 1.3. Because of this pleiotropy, we hypothesized that TF activity could be estimated from a relatively low number of reads per transcriptional target, as long as there were multiple targets from which to sample. This would allow the possibility, at least in theory, of coupling the reduction in sequencing

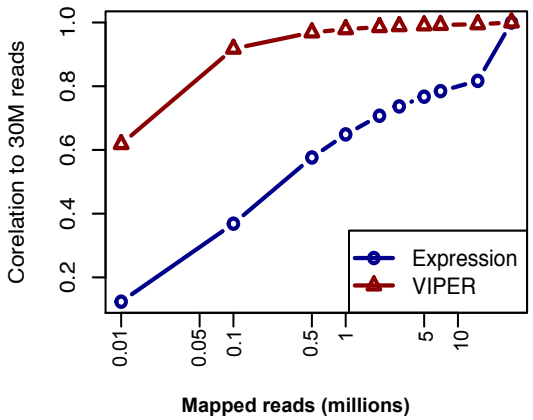

Figure 1.4: VIPER retains strong positive correlation to a larger network across a range of read depths. Alvarez, 2012.

depth per sample to an increase in the multiplexing of samples in an expression library. Because each TF, on whose activity VIPER reports, is inferred from multiple transcripts, we hypothesized that fewer total reads would be needed per experiment, to successfully run VIPER. One of the difficulties in working with low-depth RNA-Seq data is that the strength of a gene expression signature decays rapidly with reduced depth [87]. To test VIPER's sensitivity to sequencing depth, we interrogated 100 breast cancer RNA-Seq experiments from the TCGA database. Reads from each experiment were sampled, starting at 30 million (30M) and going down to only 10,000. Gene expression profiles and VIPER-inferred TF activity profiles were computed at each sampling and the correlation to the full 30M reads was measured. Figure 1.4 shows the average correlation between each down-sampling experiment and the corresponding sample at the full 30M reads. This demonstrates that although the gene expression signatures are quickly degraded, the VIPER-inferred TF activity signatures remain highly conserved across a range of sequencing depths, only significantly falling off after approximately 500,000 reads. This suggested that VIPER could accurately infer network characteristics at even very low read depths, thereby increasing the number experiments that could be multiplexed into a single sequencing run.

In another early test of VIPER's efficacy, data from a single-cell glioblastoma

(GBM) RNA-Seq study were used to compare the relationship between the detection of TF expression and inferred activity. The TFs identified in the study could be classified into three groups; those which were detected at the level of mRNA in all samples, those which were detected in some samples and those, whose mRNA transcripts were not detected in any of the samples. As can be seen in Fig 1.5, VIPER was able to infer the relative activity of TFs, even when their coding mRNA fell below the detection limit.

Efforts have been made in the past to increase the number of transcription libraries that can be sequenced simultaneously or to in other ways reduce the number of needed inputs for network reconstruction. In the former case, transcriptome library preparation methods have been developed that make use of molecular barcodes to maintain separation between pooled samples [37], [77], [47]. In the latter case, initiatives like the LINCS program have sought to find a minimal set of genes, whose differential expression is informative of diverse cellular states. These 'reduced representation' gene expression profiles, however, have proven to be ill-suited for network approaches, which tend to rely on a greater number of inputs. Despite the number of barcoding and pooling strategies available for building multiplexed transcription libraries, all of them rest on the assumption that a high sequencing depth of roughly 30M reads will be needed for network deconvolution. Many studies that would make use of highly multiplexed library sequencing would benefit most from being able to sequence a very large number of samples, covering multiple experimental conditions, such as drug screens, in which it is important to sample across multiple time points and drug concentrations. Performing this level of sequencing at high depth remains prohibitively expensive.

1.5 Pooled Library Amplification for Transcriptome Expression Sequencing

Here, I present a method for preparing transcriptome libraries for reduced depth sequencing and network analysis. I call this method Pooled Library Amplification for Transcriptome Expression Sequencing, or PLATE-Seq. The fundamental point of PLATE-Seq is that it is designed specifically to be incorporated into network biology approaches that

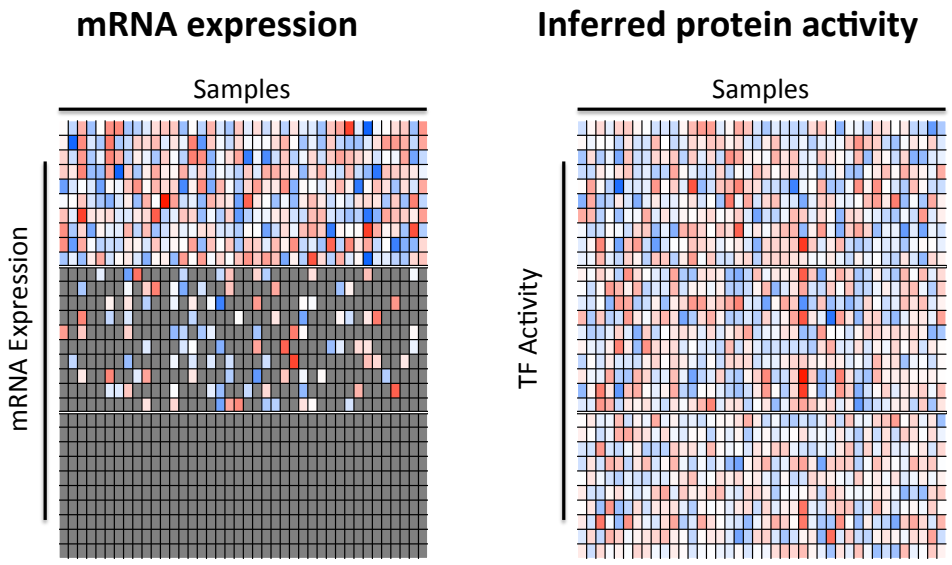

Figure 1.5: VIPER detects TF activity even when the expression for that TF could not be detected. Columns are samples, rows are TFs. Red indicates increased expression or activity, blue indicates decreased expression or activity and gray indicates that the expression of the gene in question was not detected. Alvarez, 2013.

operate on gene expression profiles. PLATE-Seq makes use of both sample-specific and molecular barcoding strategies and incorporates high-throughput liquid handling robotics to dramatically drive down the per-sample cost of expression library preparation. The gene expression profile (GEP) specificity of PLATE-Seq stems from the fact that this method does not sequence over the entire gene body, but rather adopts a poly-A tail capture technique. It furthermore relies on reduced PCR amplification to keep captured mRNA abundances closer to the linear expression range, in contrast to established techniques such as Illumina's TruSeq protocol, in which amplification bias is a serious concern. While these factors make PLATE-Seq inappropriate for studies seeking out mutations in specific genes, or alternative splicing events, it makes for a highly efficient gene expression profiling technique with substantial potential applications as a diagnostic tool.

Chapter 2

Methods

The PLATE-Seq protocol is designed to reduce the overall number of steps involved in library preparation, both as part of its goal of reducing the cost of library preparation and to minimize the amount of manual labor involved, thereby limiting the opportunity for human error and the considerable variations in experimental inputs that stem from sources such as pipetting error and minor variations in incubation lengths and interstep temporal variations.

Briefly, the protocol consists of seeding a 96-well plate with cells, administering experimental treatments and lysing the cells at the appropriate time. mRNA is isolated from the cell lysates, reverse transcribed with bar-coded oligo(dT) primers and pooled. All steps prior to pooling are performed on a liquid handling robot to reduce the occurrence of batch effects. Second strand synthesis is performed on the pool and if desired, using a second set of bar-coded random primers for future correction of PCR bias. The library is then split into several dilutions and amplified with 12 or fewer PCR cycles. The dilution that results in the minimal amount of library needed for loading into the sequencer is selected for sequencing, as this is the library, whose transcript abundances are most likely to be closest to the linear range of expression. Sequencing is carried out in the Illumina NextSeq 500. The outline of this process is sketched in Fig 2.1

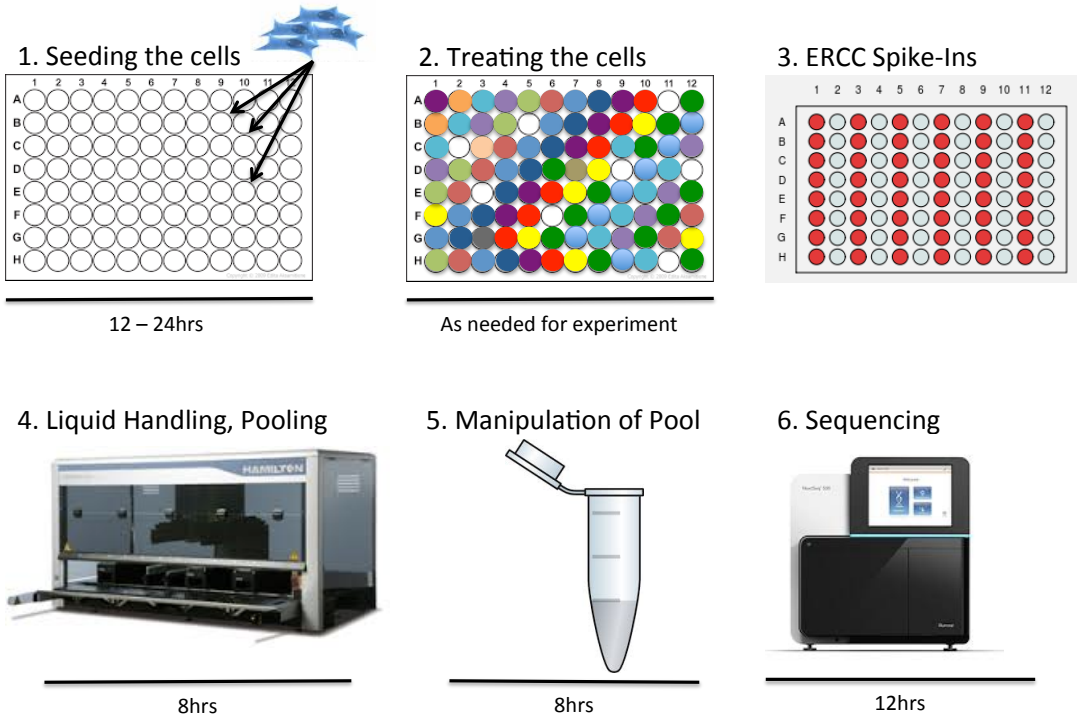

Figure 2.1: Outline of the PLATE-Seq experimental workflow

2.1 Detailed Steps of the PLATE-Seq Protocol

2.1.1 Tissue Culture

Currently, the protocol begins with seeding cells into a 96-well plate. Efforts are underway to adapt the protocol to the higher throughput setting of 384-well plates. Cells must be seeded at a concentration that allows them to grow for enough time to recover from the shock of having been split and seeded, which may vary considerably for the cell type in question. As of this writing, we have had steady success plating a variety of immortalized cancer cell lines, but have experienced greater challenges with cells of nervous tissue origin. Initial cell density must also take into account the amount of time needed to complete the experiment without overcrowding, which is a condition associated with genetic and biochemical changes that can be challenging to distinguish from the changes induced by experimental treatments [96], [57], [30]. These parameters must generally be determined empirically.

2.1.2 mRNA Isolation

The first step in RNA isolation is to lyse the cells containing the RNA of interest without causing undue RNA degradation. The steps of PLATE-Seq, from mRNA isolation until pooling of the samples, are fully automated, using a robotic protocol. The automation facility at the Columbia University Medical Campus uses a Hamilton MicroLab STAR liquid handling robot, which was used for all the automated experiments detailed herein. Lysis is performed using a buffer consisting of 99% TCL (Qiagen) and 1% beta-mercaptoethanol (βME). This buffer hypotonically lyses the cells, while the TCL, consisting predominantly of guanidine thiocyanate, inhibits the action of RNAses present in the cellular lysates. The minimal volume of lysis solution that was found to be needed to effectively lyse cells in the 96-well plate format was 30μL. Prior to lysis, cells were washed twice with PBS, to remove the culture media. Cells incubated in the lysis buffer for 5 minutes, at room temperature and the lysates were vigorously pipetted to ensure efficient cell lysis and cleared by gentle pulse centrifugation prior to being transferred to an mRNA capture plate.

The mRNA capture plate is a 96-well plate, wherein the sides of each well are coated

with plate-bound oligo(dT). The oligo(dT) efficiently anneals to the poly-A tails of the mRNA molecules. The mRNA is eluted from these wells following the manufacturerâ^{TM}s protocol and transferred to a fresh 96-well PCR plate, whereupon the ERCC spike-in controls and bar-coded, Illumina adapter-linked primers are added. 1μL of a 1:100 dilution of either of the two ERCC spike-in mixes are added to alternating wells in order to quantify the amount of variation in mRNA yields between wells and to later assess for the occurrence of cross-talk between wells. 3μL of 100μM bar-coded, adapter-linked primers are added. The plate is then heated to 94C for 2 minutes to fragment the mRNA, thereby destroying much secondary structure that may interfere with primer annealing and reducing the range over which mRNA molecules in the library may vary. This is an important consideration for PCR amplification, in which shorter molecules may out-compete longer ones for relative abundance in the library, which reduces overall library complexity. One critical consideration at this step is to transfer the freshly fragmented mRNA directly and immediately to ice in order to allow the bar-coded primers to anneal to the poly-A tails of the mRNA fragments faster than the mRNA can refold into secondary structures that would impede primer annealing.

2.1.3 Reverse Transcription Library Pooling

Reverse transcription is then performed using Protoscript II reverse transcriptase (NEB). Reverse transcription proceeds for 120 minutes at 42C, after which the reverse transcriptase (RT) is heat inactivated for 20 minutes at 65C, to limit the possibility of remaining active RT priming off of a bar-coded primer from one well annealed to a transcript from a different well. TO further control for the incidence of cross-talk, remaining primer is removed by digestion with ExoI for 1 hour following reverse transcription. Because ExoI can also degrade the ssDNA of the library, it is inactivated with 20μL of a 1:1 mixture of 1M NaOH and 0.5M EDTA for 15 minutes, after which samples are pooled together. Finally, 80μL of 12M HCL is added to neutralize solution. At this point, the samples from separate wells are pooled together and can be treated as a single sample for all downstream steps.

2.1.4 Concentration of Pooled Library

Due to constraints imposed by the robotic liquid handling system, the pooled volume is larger than needed for the downstream steps. Although the first action post-pooling is to clean the library of its ExoI neutralization buffer and concentrate it to a smaller volume, the amount of hands-on time required to concentrate it from the 40mL that the robotics facility delivers to the 15μL required for the rest of the protocol is significant without access to a vacuum spinner capable of handling 50mL conical tubes. For this reason, only an aliquot of the pooled library is concentrated, while the remainder is stored at -80C, in case more is needed later, for example, in the case of having to repeat a sequencing run. As the cell culture and liquid handling components of the protocol are also the most time-intensive, having a reserve of pooled library serves as a practical safeguard against having to repeat an experiment from the very beginning. Library concentration is performed on silica membrane columns using the Zymo DNA Clean and Concentrate kit as per the manufacturer's "ssDNA" protocol. To further ward against contamination by residual primers, which can affect library amplification efficiency, the pooled library is purified with Ampure XP beads at a 1:1 beads-to-library ratio (by volume) and eluted in 15.0μL of nuclease-free water.

2.1.5 Second Strand Synthesis

Second strand synthesis is performed using the large Klenow fragment, which has exonuclease activity. The exonuclease activity helps to further limit cross talk by eliminating any remaining well-specific primers that may have carried over through the prior steps. 1.0μL of a 10mM dNTP mix and 1.0μL of a 100μM random bar code primer mix are added to the library, which is then incubated at 70C for 2 minutes and immediately transferred to ice, to incubate for another 5 minutes. 2.0μL of Buffer 2 (NEB) and 1.0μL of DNA polymerase I are then added to the library and second strand synthesis is allowed to proceed by incubating the entire mixture at room temperature for 30 minutes. The resulting double-stranded cDNA is purified with Ampure XP beads at a 1:1 ratio, to remove all residual primers and finally eluted in 20μL of nuclease-free water.

2.1.6 Amplification

Because of the reduced sequencing depth, it is critical to minimize PCR amplification bias. For PCR amplification, three dilutions of the library are made; a 1:2.5 dilution of the original solution and a 1:2 dilution of the 1:5 dilution. 2.5μL of the original library and each dilution were mixed with 22.5μL of a PCR reaction mix for final library dilutions of 1:10, 1:25 and 1:50. The libraries were amplified with 12 PCR cycles. The products were purified twice with Ampure XP beads at a 1:1 ratio, eluted in 12.0μL of nuclease-free water and aliquots were taken for quantification by qubit and analysis by Bioanalyzer.

2.1.7 Sequencing

Sequencing for PLATE-Seq experiments is performed by paired-end sequencing on Illumina's NextSeq 500 v2 (high throughput) platform. The NextSeq 500 is the current model of Illumina's reversible-terminator sequencing-by-synthesis (SBS) technology. Reversible termination refers to the use of dNTPs, modified to contain a fluorescently-labeled terminator molecule, which inhibits further polymerization after incorporation into a growing strand. This prevents the addition of multiple nucleotides in a single sequencing round, which would otherwise significantly increase the sequencing error rate. Sequencing is carried out simultaneously on millions of template cDNA fragments bound to the surface of the flow cell. The flow cell is coated with adapter sequences, which anneal to the template cDNA molecules through complementary sequence binding. The initial input templates are amplified to produce clonal template clusters, which are vital for quality control purposes. The presence of both forward and reverse strands in the sequencing reaction provides a check against sequencing artifacts. If one strand is found to have an odd sequence, then it can be checked against its paired strand to confirm whether or not the complement contains the same oddity. Clonal clusters also amplify the signal from each fluorescent dNTP used in the sequencing reaction, as a single fluorescent label falls below the limit of sensitivity of the deviceâ$^{\text{TM}}$s cameras. Following clonal amplification, the sequencing reaction proceeds in cycles, with a single nucleotide added at each cycle. After the addition of each nucleotide, an image is taken of the fluorescence pattern across the flow cell and the terminators on each nucleotide are cleaved to allow incorporation of a new nucleotide in the subsequent

cycle. The incorporation of a single base per cycle allows the sequencing reaction to produce a set of reads of uniform length. The reaction proceeds for a defined number of cycles, after which the raw data is sent for primary read alignment and downstream processing.

2.2 Data Analysis

2.2.1 Sequence Alignment

The primary processing of the raw reads is a fully automated process, in which the reads generated during sequencing are mapped to the appropriate genome using the STAR Aligner [23]. The output of this process is a text file containing the gene counts for each sample (bar code) in the library.

2.2.2 Normalization and Comparison of Gene Expression Profiles

Data normalization is first performed by variance stabilization of the raw counts using DESeq2, an update of the well-established DESeq package [1], [54], [32]. The theory behind DESeq and DESeq2 is that HTS experiments frequently suffer from a small sample size, likely because the cost of performing sequencing at high depth restricts the number of samples that can be sequenced. These small sample sizes lead to large uncertainties of intragroup variance estimates and a resulting lack of statistical power. DESeq and DESeq2 meet this challenge by modeling the dependence of the amount of variation, or the dispersion, on the average expression strength across all samples. Dispersion refers to the variability in counts between replicate treatments and proper estimation of this variation is critical to accurately infer differential gene expression. DESeq2 makes the assumption that genes of similar average expression strength will display similar dispersion. The algorithm estimates dispersion by first estimating variability on a per-gene basis, then fitting these estimates to a smooth curve to produce an accurate estimation of expected dispersion values for each gene. It then shrinks the gene wise dispersion estimates towards the values of the fit curve. These shrunken values are the final dispersion estimates. The strength of each gene's shrinkage factor depends on both how close true dispersion values are to the fit and on the degrees of freedom in the dataset. The more samples there are, the less strong the

effect of shrinkage becomes.

2.2.3 The VIPER Algorithm

The VIPER algorithm tests for the enrichment of TF-specific regulons in gene expression signatures (Alvarez, et al., in review). Regulon enrichment is computed through an algorithm called the analytic rank-based enrichment analysis (aREA). aREA detects shifts in regulon enrichment by comparing the positions of regulon genes to rank-sorted gene expression profiles and assigning to each regulon gene an enrichment score (ES). aREA first computes the ES for regulon genes by rank-sorting them based on their absolute expression fold change, invariant to the direction of their changes in expression. In this way, the genes with the greatest changes in expression in the experimental condition are easily identified. In the next step, the algorithm re-orders the list of differentially expressed genes, based on the direction of regulation between the gene and its regulator. The results of these two calculations are integrated and the contributions of each are weighted based on the results of another part of the algorithm, called the Mode of Regulation (MoR). The MoR describes the direction of interaction between a regulator and a target. It is based on the Spearman correlation coefficient (SCC) between a regulator and the expression of a transcriptional target, as computed from the dataset used to reverse engineer the network. Finally, aREA calculates the statistical significance for each ES through comparison to a null model that is generated either by a random uniform permutation of the samples or by a random uniform permutation of the genes in the dataset, should there be too few samples to generate sufficient statistical power.

2.2.4 ARACNe

Because of the central role that ARACNe-inferred networks play in VIPER analysis and in more generally in the current PLATE-Seq experimental pipeline, it is worth describing the mechanics of this algorithm in greater detail. ARACNe reverse engineers biological networks in two steps. Broadly, it first estimates the pairwise mutual information between all genes in a gene expression profile and filters these interactions based on a threshold MI, I_0, computed for a specific p-value, p_0, in the null-hypothesis of two independent genes.

This step is essentially equivalent to the Relevance Networks method [9], and, as such, suffers from the same limitations. In particular, genes separated by one or more intermediaries may be highly co-regulated without implying a direct physical interaction.

The second step in ARACNe is designed to account, at least in part, for the effects of indirect regulation. In this step, ARACNe removes the majority of candidate indirect interactions through an application of the data processing inequality (DPI) [18]. The DPI argues that if two genes, g1 and g3, interact only through a third gene, g2, and no alternative path exists between g1 and g3, then the information between g1 and g3 cannot be greater than that which is estimated between either of the other two pairs. Formally,

$$I(g1, g3) \leq min[I(g1, g2); I(g2, g3)]. \tag{2.1}$$

The implication is that the smallest of the three MI values in a triplet implies an indirect interaction. ARACNe examines each triplet set of interactions in which the MI for each interaction is greater than I_0 and culls the edge with the smallest value. Because each triplet is analyzed independently of whether its edges were previously marked for removal as part of another triplet, the network that ARACNe reconstructs is independent of the order in which triplets are examined.

Given ARACNe's focus on pairwise interaction, it is not well-suited to identifying higher-order interactions between mutually independent genes, or those for which $I_i, j < I_0$. It would also fail to recover pairwise interactions, wherein the effect of a direct interaction is cancelled out by indirect actions transmitted through other genes. However, such precise cancellation, particularly if carried out systematically, can be deemed biologically unrealistic.

Chapter 3

Development

The PLATE-Seq protocol has gone through several iterations to arrive in its current, and still evolving, state. PLATE-Seq was originally proposed by Peter Sims as a potential protocol to address one of Andrea Califano's research goals. The Califano lab has grants for several projects that require or would significantly benefit from considerable amounts of sequencing. These include efforts to construct libraries of transcriptome-wide drug profiles in multiple cellular contexts and using transcriptomics to search for master regulators of individual drug response through testing multiple drug treatments against patient-derived tumor tissue. It became my task to develop and test the protocol and to shephard it towards something that could be used reliably at high scale.

All iterations of the PLATE-Seq protocol follow the essential steps involved in RNA purification and library preparation. The key differences are early pooling and reduced depth sequencing. The latter is only rendered useful thanks to network biology approaches like the VIPER algorithm that boost the information we gain from low-depth sequencing results by incorporating prior gene regulatory knowledge into the analysis.

Our goal was to able to perform PLATE-Seq on multiple 96-well plates. The plan is to eventually adapt the protocol to 384-well plates to further increase the scale of the procedure, but this will be a project for future lab members. We knew going in that developing a reliable and replicable protocol for 96-well plates would be an already significant challenge. Peter Sims and I developed a plan to develop the protocol in stages. After working out an outline for the protocol, I would test each step in a single sample setting, looking for any

weaknesses in each procedure that was to be performed. Specifically, we wanted to show that we could efficiently purify mRNA while removing any rRNA and gDNA contamination. We needed to know the size range of the libraries that came from our protocol, to be able to control the length distribution and we needed to know the minimal amount of PCR amplification that we could use, to minimize PCR-induced length bias without sacrificing library complexity.

After working out any bugs in the single-sample setting, I would then scale the protocol up to a higher throughput setting of 48 wells. Although anything that works in 48 wells should work just as well in 96, we chose not to assume success and to scale the procedure up to just half a plate first. Doing so would also reduce the reagent cost involved in performing an experiment that may contain unforeseen bugs.

3.1 Testing Protocol on Single Samples

The first protocol involved single samples of total RNA, purified from MCF7 cels. I performed mRNA purification from this starting material using a streptavidin-biotin pull-down. Immediately following the mRNA capture and prior to the bead pull-down, I split the sample into two aliquots, one of which would be treated with exonuclease I (exoI) and the other of which would not, to determine whether or not digestion by exoI was needed to remove biotinylated transcripts from streptavidin beads. In the first trials of the protocol, I captured mRNA by annealing biotin-conjugated oligo(dT) to the poly-A tails of mRNA and performed reverse transcription using bar-coded and Illumina adapter-linked random primers. I next used Zymogen's RNA clean-up kit to remove DNA contamination, following the manufacturer protocol. I then purified the captured mRNA with streptavidin, incubating each sample on a rotisserie for one hour at 4 degrees. Second strand synthesis was then performed using random hexamer barcodes and klenow(exo-) and the resulting double-stranded libraries were then amplified with the Phusion DNA polymerase as in later protocols.

Because the protocol at this point was in the very early stages of development, care was taken to measure the output of every step. At each washing step (the the RNA clean-up

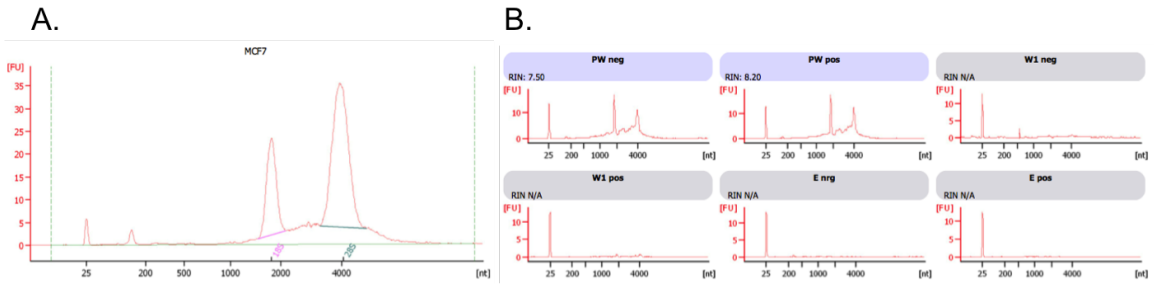

Figure 3.1: A. Total RNA purification from MCF7 cells. The two prominent peaks show rRNA. B. Results of bead pulldown and washing. PW = pre-wash, W1 = 1st wash, E = eluate. Although the library concentration at these steps is still too small to see on the bioanalyzer, the protocol steps effectively remove rRNA.

and after the bead pull-down), aliquots of the library were removed prior to the wash, after a first wash and from the eluate, to assay for the presence of potential library components in anything but the eluate.

Results at these stages were encouraging, showing peaks tracking to the expected sizes for ribosomal RNA in the pre-wash samples only, following bead pull-down, as assayed on the bioanalyzer. At this point in the protocol, the amount of unamplified library remained below the limit of detection of the bioanalyzer, making an apparently blank trace a potentially positive result (Fig 3.1).

At this stage, we knew that minimizing the degree of PCR amplification bias in the libraries would be critical to success, but didn't know how many cycles of PCR we should use. To get an empirical feeling for how amplification affected the library, I split the double-stranded, non-amplified library into five separate aliquots and amplified each aliquot with a different number of cycles, covering several orders of magnitude of amplification. Initially, I used 8, 12, 16, 20 and 30 PCR cycles. I assessed relative amplification and determined library length by running each amplified aliquot through a 1.5% agarose gel. This first attempt showed only a smear in both lanes of the 30 cycle samples, which I reasoned would represent strong overamplification. I repeated this procedure and found a smear in only the exoI+ lane at 16 cycles that ranged the length of the size ladder, with a denser bulge in

the range of approximately 300bp - 600bp (Fig 3.2). No smear was visible in the exoI- lane. This suggested that exoI was, in fact, needed to free biotinylated transcripts from beads following purification. I controlled the library length by excising the smear approximately between 230bp and 600bp and then measured library concentration by qPCR.

Buoyed by the results of the single-sample experiment, I adapted the protocol to the higher-scale setting of one half of a 96-well plate. The first step was to optimize the number of cells per well. Initially, I seeded 48 wells with 5,000 MCF7 cells per well, but was unable to see evidence of a library in the gel following amplification. After discussing this results with Peter, we decided that rather than expending resources on a blind gel purification and qPCR, I would repeat the procedure, doubling the number of cells per well. This could later be attenuated, but we needed a library that we could see, in order to track the progress of the current protocol. I seeded 10,000 MCF7 cells per well in the following experiment. However, bioanalyzer traces showed that the starting total RNA may have been heavily degraded and the gel following the PCR showed smears that were heavily concentrated below 600bp, further indicating that there were likely structural problems related to transcript quality in this sample. I repeated the experiment, using more superaseIN in the lysis buffer, to counter any unwanted RNA degradation. I amplified aliquots of the library using 12, 20 and 24 cycles of PCR, then gel purified and quantified the resulting libraries.

After convincing ourselves of the technical success of the 48 well trial, I performed three trials using full 96-well plates. The first was a run of just the pre-sequencing steps, the second was a full sequencing run performed to assay for cross-talk between wells and the third was an experiment with drug treatments. In the first case, all steps appeared successful, with the final result of a 20nM library; more than enough to be sent for sequencing. It was at this point, that the idea came to load the wells of a plate in such a way that we could assess the presence of any cross-talk between wells. I forget who thought of it first, but after it was brought up, Peter and I decided that rather than send the library that I had just built for sequencing, I would set up another run, this time with alternate wells seeded with cells that could be separated by barcodes post-sequencing. For this experiment, I chose to use two cell lines that had been given to me by the da Silva Lab. One was an MCF10A line that expressed shSTAT3 and and the other was an MCF10A line that contained the empty

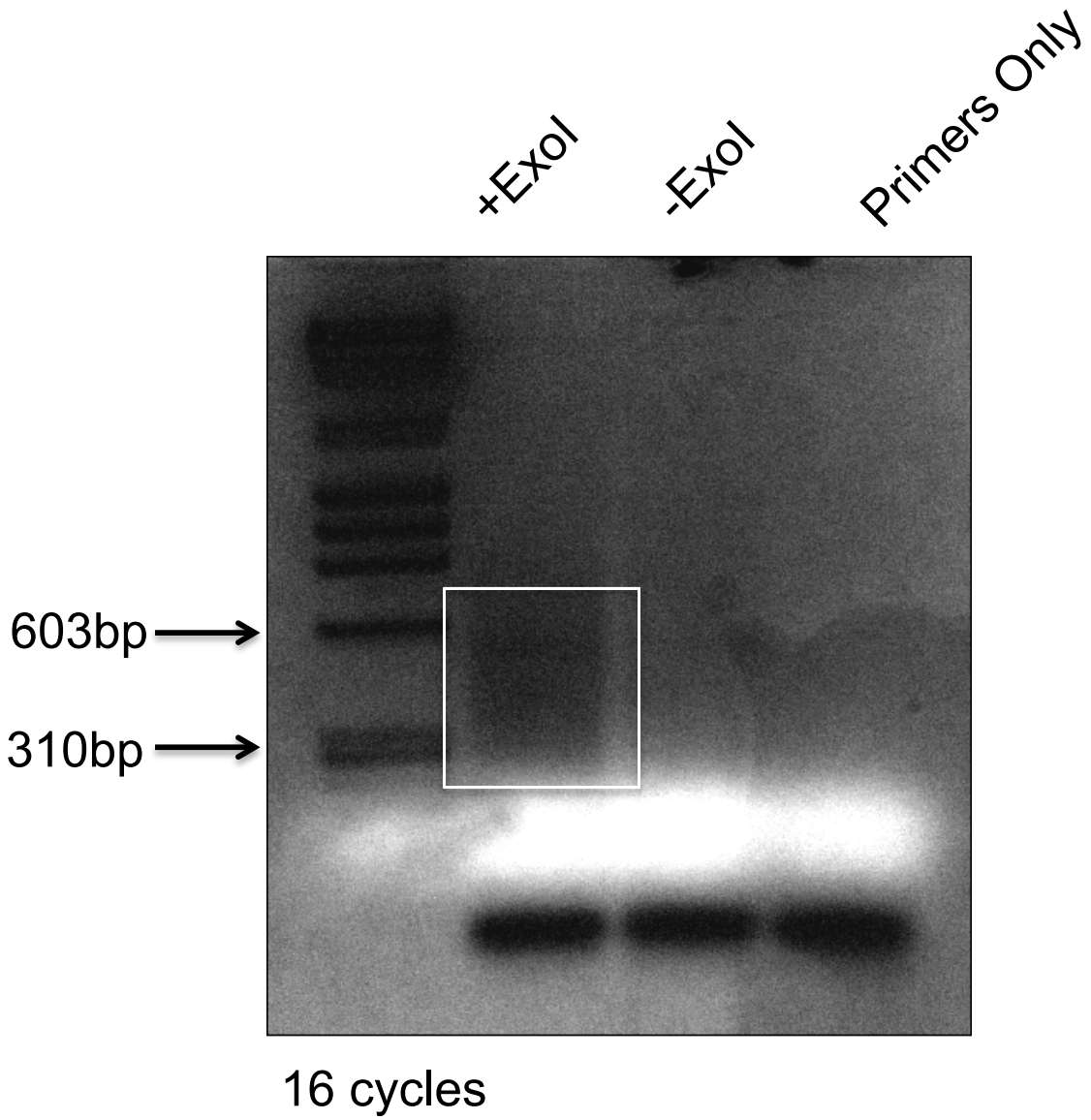

Figure 3.2

shSTAT3 vector. since MCF10A cells normally express STAT3 at high levels, I would be able to use STAT3 as a marker for cross talk by mapping bar-coded sequences to each well. Unfortunately, I could only detect STAT3 in 28% of all samples in the library. I checked STAT3 expression in each cell line and found that it was undetectable in both. The lesson learned in this experiment was to validate all claims concerning reagents prior to using them.

I repeated this experiment, alternating columns of a 96-well plate with unmodified MCF7 cells and GFP-expressing MCF10A cells. I verified GFP expression by microscopy. This showed that all bar codes were well-represented in the resulting library, that reads mapping to the GFP vector were detected in 94% of the expected wells and that 99.96% of the reads that uniquely mapped to GFP were associated with wells to which GFP+ cells were added (Fig 3.3). The 0.04% of GFP reads mapping to other wells were traced to a single bar code in a single well. This demonstrated a sufficiently high level of well-to-well fidelity that we decided to design an experiment to further test the quality of information that we could obtain through PLATE-Seq.

We next designed an experiment involving drug perturbations, to evaluate our ability to obtain clear and differentiable gene expression profiles and VIPER-inferred protein activity profiles using PLATE-Seq, this being the primary motivation for developing the technique. This experiment consisted of a set of seven drug treatments at two concentrations and two points, with three biological replicates per treatment. For this version of the protocol, Peter and I decided to add a phosphatase digestion step, following the exonuclease digestion, which would strongly limit any residual action by reverse transcriptase, ruling that out as a potential source of the cross-talk seen previously. The results of this experiment showed that treatment replicates tended to be well-correlated, both at the level of gene expression and of VIPER-inferred protein activity, as measured by the Pearson correlation between either gene expression or protein activity signatures. A drawback to this analysis, however, was that we did not see strong differences between signatures from diverse treatments. While discussing ways to overcome this, we were approached by collaborators, who were interested in the potential benefits of PLATE-Seq and in doing experiments with us, while we developed the technique further. These experiments are discussed in the following

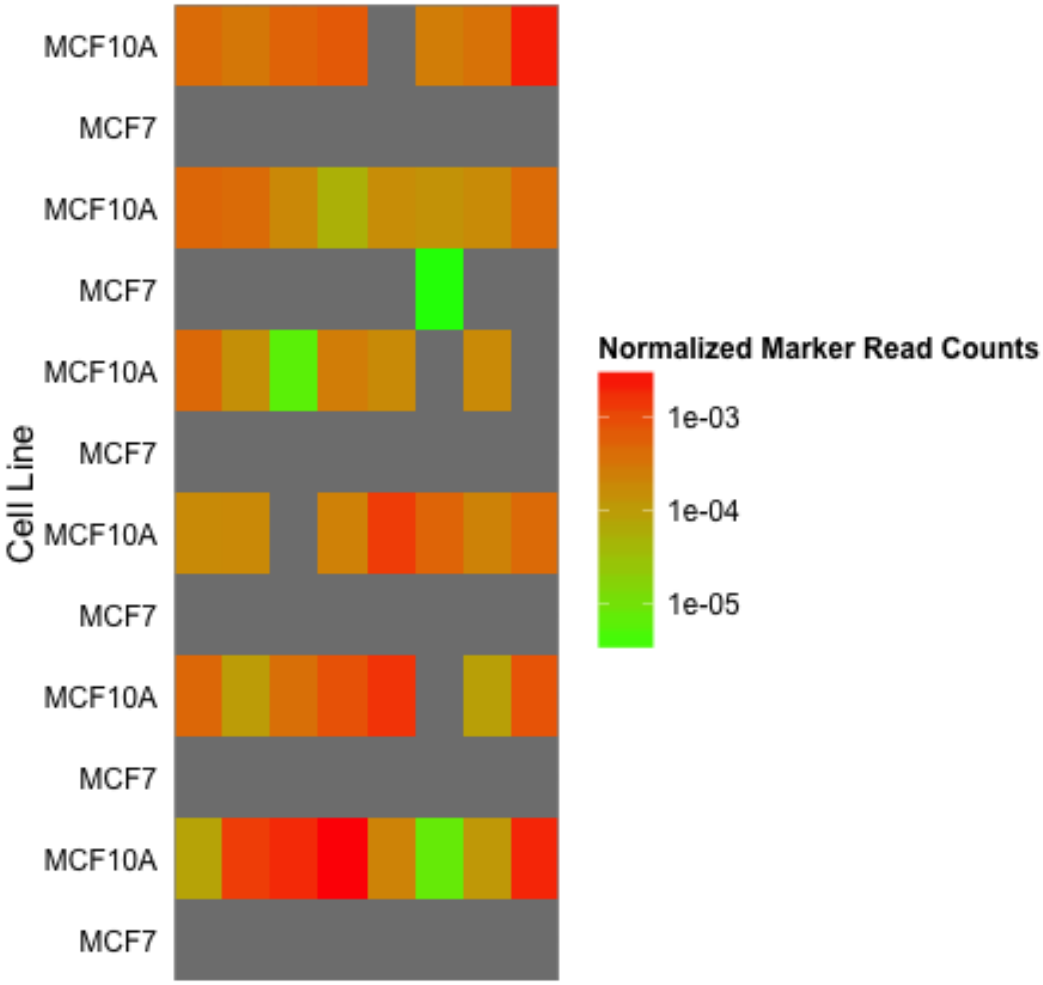

Figure 3.3: GFP marker representation across the 96-well plate.

chapter.

To complete the discussion of how I developed the PLATE-Seq protocol, I'll end by briefly mentioning the experiments that I carried out in parallel with those detailed in the next chapter.

One interesting collaboration that was proposed to us by Dr. Serge Przedborski involved adapting PLATE-Seq for use with motor neurons. The key challenge in this set of experiments was the cell culture. Unlike immortalized cell lines, motor neurons require significantly more space between cells in order to grow in a healthy manner and so cannot be grown to confluency. The must also be cultured for a longer period of time than immortalized cells, which makes them more prone to culture phenomena such as the edge effect, in which cells grown in the peripheral wells in a multiwell plate lose a greater proportion of their culture media to evaporation, negatively impacting their health. These effects gave me fewer cells to work with than in past experiments. This, in turn, caused me to use to a greater number of PCR cycles when amplifying the library, which lead to critical loss of library complexity due to amplification bias. It is important to note that at this stage in protocol development, I was still heat fracturing mRNA to homogenize read lengths prior to reverse transcription. Later versions of the PLATE-Seq protocol incorporated a plate-based 3-prime mRNA capture, followed by first strand synthesis by priming directly off of the poly-A tail, which significantly improved mRNA capture efficiency and library complexity, as detailed in the Methods chapter. Although I am no longer involved in the motor neuron project, it is my understanding that a much greater degree of success has been found by employing the most current protocol.

In a further effort to reduce opportunities for cross-talk, I experimented with comparing the effects of treating wells with exoI and recombinant shrimp alkaline phosphatase (rSAP), which removes the phosphate needed by RT to carry out its function, to heat-killing RT by adding a 20-minute, 65C step at the end of the reverse transcription thermocycler protocol. This experiment showed that heat killing RT was equivalent to the exoI + rSAP digestion, which allowed us to save on reagent cost by excluding rSAP and limiting the amount of exoI used in the protocol.

One strategy that we briefly pursued was the incorporation of a terminal transferase

step into the protocol. Terminal transferase adds adenine nucleotides to the 3' end of ssDNA. Peter and I reasoned that this would reduce reagent costs and improve second strand synthesis efficiency by allowing second strand priming to take place on a single sequence, shared by all reads. Although this worked, we shortly afterwards came up with the idea of using the oligo(dT)-coated plate for mRNA capture, followed by thermal fragmentation of the mRNA molecules, which was both more efficient and less laborious, making it better-suited for the eventual current working protocol, as detailed in the chapter on methods.

Chapter 4

Results

4.1 Initial Development

In developing PLATE-Seq, our primary technical challenge has always been to demonstrate that results obtained through PLATE-Seq and analyzed by VIPER are functionally equivalent to those obtained through standard expression library preparation methods and high-depth sequencing. To this end, we have performed several comparative experiments to test PLATE-Seq's efficacy in various conditions.

4.2 Testing PLATE-Seq Results Against a Matched Dataset of Previous Sequencing Results

As an early test of the PLATE-Seq method, we repeated a previous drug screen that had been performed as part of a project involving neuroendocrine tumors. The previous screen had been performed using the TruSeq expression library preparation method and libraries from that project had been sequenced to a depth of 30M reads on the HiSeq 2000 (Illumina).

4.2.1 Experimental Set-Up

Two 96-well plates were seeded with 30,000 H-STS cells 16hrs prior to drug treatment. To test both the reproducibility of PLATE-Seq samples and to get an estimate

	1–5	6–10	11	12
A	AZD8055 - 20uM	MK-2206 - 6.84uM		
B	AZD8055 - 2uM	MK-2206 - 0.684uM		
C	Belinostat - 0.0325uM	Tivatinib - 0.709uM		
D	Belinostat - 0.00325uM	Tivatinib - 0.0709uM		
E	Entinostat - 6.84uM	Topotecan - 0.0138uM		
F	Entinostat - 0.684uM	Topotecan - 0.00138uM		
G	Imatinib - 20uM	YK-4-279 - 0.65uM		
H	Imatinib - 2uM	YK-4-279 - 0.065uM	DMSO	Empty

Figure 4.1: Plate layout for comparison to NET drug screen.

of library complexity at low cell counts and sequencing depths, we alloted five biological replicates for each treatment. Drugs were administered at the concentrations indicated in Fig 4.1 and incubated for 6 and 24hrs, to match the conditions of the original screen.

4.2.2 Experimental Protocol

At each time point, growth media was removed and cells were washed twice in sterile PBS before being lysed by 15 minute room temperature incubation (27C) in a hypotonic buffer containing DNase (Turbo DNase, ThermoFisher). Cells were then transferred to a 96 well mRNA capture plate, in which the walls of each well are coated with bound poly-T sequences to capture the mRNA at its poly-A tail (TurboCapture mRNA plates, Qiagen), following the manufacturer's protocol. mRNA was eluted from the capture plate and transferred to a fresh 96-well plate, in which it was fragmented by heating at 95C for 6 minutes, followed by immediate incubation on ice. Reverse transcription (RT) was carried out in this same plate, during which well-specific bar codes were introduced, as a component of the RT primers. The resulting single-stranded cDNA (ssDNA) was pooled and concentrated through silica-based membrane columns (Zymo Clean and Concentrate kit, DNA protocol) and further purified with Ampure XP beads, to remove any residual primers. Second strand synthesis was carried out by incubation with 5U of klenow fragment (exo-), for 10 minutes at 25C, followed by 50 minutes at 37C. The newly double-stranded cDNA (ds-cDNA) was then purified with Ampure XP beads, as before and split into several aliquots and amplified by PCR. Each aliquot was amplified with a different number of PCR cycles and the library that showed the minimum amount of material necessary for sequencing was used. Using the least amplified library helped to keep the amount of amplification closer to the linear range,

thereby limiting the amount of PCR bias expected to be found in the resulting library. Length-adjusted library concentrations were calculated based on measurements from an Agilent 2100 Bioanalyzer, to establish mean library fragment length and a Qubit fluorometer (Invitrogen), to measure the library concentration by mass. Libraries were submitted to the Sulzberger Genome Center for sequencing on the HiSeq 2000 platform (Illumina).

4.2.3 Results

The sequencing data obtained after library preparation with PLATE-Seq was generally of very high quality. The low proportion of reads mapping to rRNA sequences indicated an efficient mRNA purification (Fig 4.2A). To measure the distribution of our random primer annealing during second strand synthesis, we mapped the distribution of reads covering each nucleotide position of all identified genes and compared this to a previous RNA-Seq study involving a library that was sequenced to a depth of 30M reads per sample. As shown in Fig 4.2B, despite the greater and expected variability in the frequency of gene body coverage in the PLATE-Seq library, full gene body coverage comparable to that of the higher-depth sequencing run is obtained. We needed a separate study for this comparison, as this analysis was not performed in the original screen and I did not have access to that data. Gene detection efficiency, or the rate at which genes were uniquely identified was measured both as a function of the number of samples in the plate and of the number of mapped reads, as shown in Fig 4.2C and D. Fig 4.2C shows the variability in identified genes between samples on the plate, with a very similar number of genes being identified in the majority of samples and only a very few outliers in either direction. We interpret this data to mean that the gene detection efficiency and therefore the mRNA purification and library amplification are fairly uniform and without obvious bias. Fig 4.2D shows that gene detection occurred very rapidly as a function of mapped reads and saturated early, decreasing rapidly after 200,000 reads, with very few new genes being identified after 500,000 reads. This data not only matched the gene detection rate of the same long-lost dataset that was used to compare gene body coverage, but closely matches the theoretical threshold for effective VIPER analysis, seen in Fig 1.4. This suggests that although VIPER functions well at low read depth, it requires a saturated gene expression library to perform optimally.

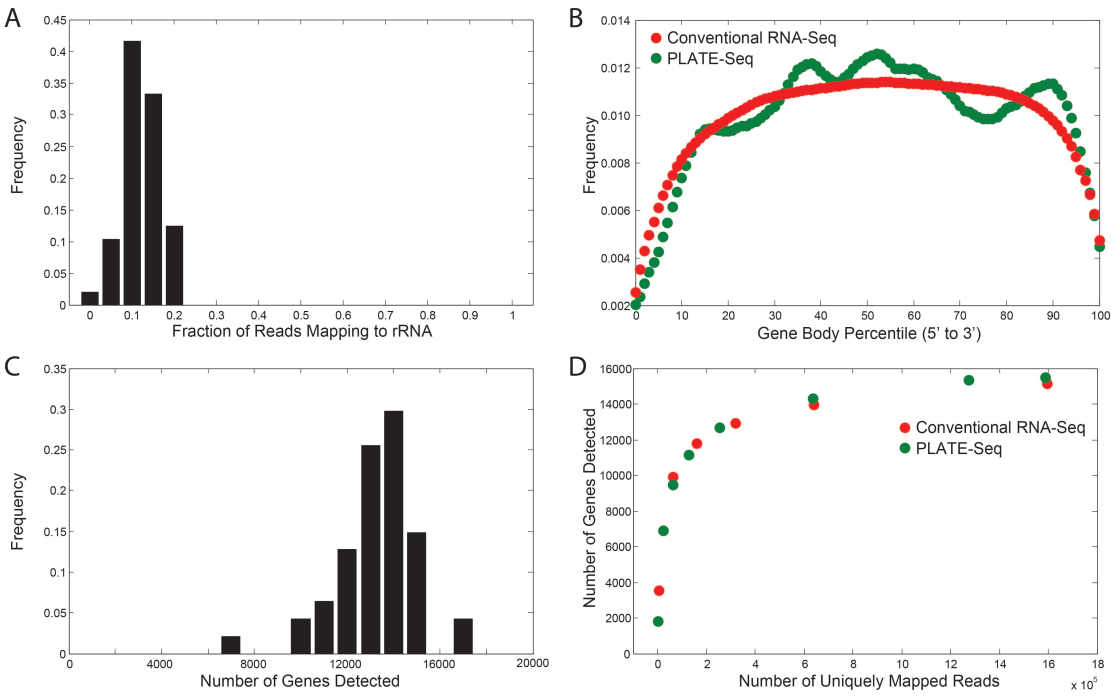

Figure 4.2: Quality control measures of the drug screen. (A) Amount of reads mapping to rRNA sequences. (B) The percentage of reads that covers each nucleotide position of all detected genes scaled to 100 bins, from 5' UTR to 3' UTR. (C) Gene detection frequency. (D) The number of uniquely identified genes as a function of the number of mapped reads.

Similarity between samples was assessed using the viperSimilarity metric of the VIPER package. The viperSimilarity compares transcription factor activity similarity between samples in the following way. For any pair of treatments A and B, it first subsets the tails of the TF activity distribution for each treatment, selecting the 50 most differentially active TFs in either tail. It then performeds GSEA, computing the enrichment of these TFs in A on the full set of TFs in B. Finally, it repeats this calculation, switching B for A. Overall, the PLATE-Seq drug treatments showed strong positive similarities to their high-depth counterparts, as seen in Fig 4.3.

One obvious exception is entinostat, which actually shows a complete reversal of signature between the two matched experiments. Although in any drug screen, one expects to see some samples fail to produce consistent or even coherent results and as long as this is not a systematic effect seen to a statistically significant degree throughout the screen, the anomalous results can be discarded and if needed, those samples can be re-screened at a later date. The complete reversal of the entinostat signature in this case proved more perplexing. At first glance, one is tempted to think that a sample label was switched. Excessive back-searching of the PLATE-Seq pipeline could not uncover any evidence of this having happened and insufficient evidence existed concerning the original drug screen to determine if such an event occurred then. Whatever happened appears likely to have occurred during the original screen, based on the similarity between TF activity signatures of PLATE-Seq-prepared entinostat and the belinostat samples. Both drugs are histone deacetylase (HDAC) inhibitors, with very similar MoA. The two clusters that appear by this analysis are one involving AZD8055 and MK-2206, both of which target the mTOR and AKT pathways and the other drugs, which for this plate, consisted of a mix of inhibitors of HDACs, kinases and topoisomerase.

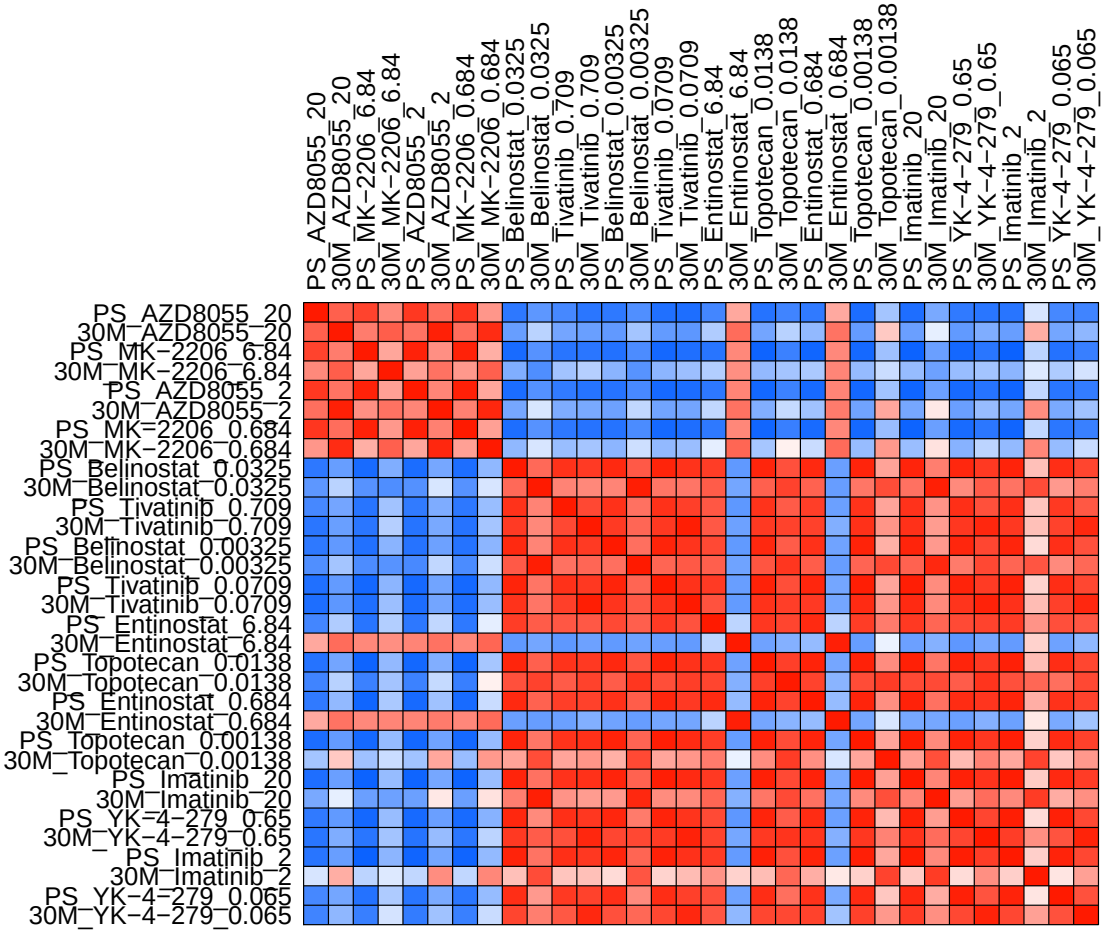

Figure 4.3: VIPER similarities of drugs prepared in separate experiments, by either PLATE-Seq or TruSeq

4.3 Using PLATE-Seq to Identify Transcription Factors That Regulate FGF-Induced Growth and Inhibition: Collaboration With Merrimack Pharmaceuticals

4.3.1 Motivation

The second test of PLATE-Seq, comparing it to established expression library preparation methods, consisted of using a targeted fibroblast growth factor receptor 2 (FGFR2) antagonist to identify candidate master regulators of FGF-induced growth and growth inhibition for use in treating squamous cell lung cancer. There are currently only limited treatment options available for squamous cell lung cancer [61]. Although numerous clinical trials are investigating the FGF pathway for actionable therapeutic potential, no approved FGF-targeted therapies currently exist. To address this, we engaged in a collaboration with Merrimack Pharmaceuticals, who were interested in developing targeted therapies to halt tumor-driven FGF-dependent angiogenesis for treating squamous cell lung cancer.

4.3.2 Fibroblast Growth Factor Receptor as a Therapeutic Target

FGF receptors are bound with varying affinity by a large number of ligands [69]. One such ligand, FGF2, is upregulated in a number of cancer settings [89], [81], [66] and binds to a subclass of FGF receptors known as the c isoforms. Cells that display FGFRc isoforms tend to be phenotypically mesenchymal, a phenotype which carries with it a decreased prognosis for long-term patient survival [90]. Merrimack had designed an FGFR inhibitor that acts by competitively binding that receptor, presumably with a higher affinity than does FGF2. They had previously tested it to show that it did, indeed, inhibit proliferation. Because of the commercial and competitive nature of this project, I do not have access to their data concerning binding affinity and the inhibition of proliferation. Merrimack was interesting in characterizing the mechanism of action of their FGFR2 inhibitor. For our part, this collaboration represented an ideal opportunity to perform technical comparisons between PLATE-Seq and TruSeq, Illumina's well-established expression library preparation method. Under our agreement with Merrimack concerning publication of results, I will

only describe the comparison of PLATE-Seq to TruSeq and the reproducibility between biological replicates seen in the PLATE-Seq prepared samples. The following section will describe the experiment that we performed for publication, which was based upon the results and modifications to the protocol that we established over the course of this project.

4.3.3 Experimental Set-Up

To compare PLATE-Seq to TruSeq and also to test PLATE-Seq's reproducibility, I profiled two squamous cancer cell lines: colo699 and H2172. The two cell lines differ in their response to FGF2 treatment. Colo699 expresses FGFR2c and as such, is sensitive to treatment with either FGF2 or the inhibitor. For this reason, it is classified as the "responder" cell line. Conversely, H2172 is a "non-responder" cell line and as such, does not express FGFR2c and should therefore be insensitive to either treatment. Although the greater and still ongoing collaboration involves many more responder/non-responder cell line pairs, my contribution consisted only of developing the PLATE-Seq protocol in the context of these two lines and performing technical comparisons to TruSeq.

The experiment consisted of treating each cell line with either FGF2 or the inhibitor at concentrations that were empirically determined to elicit maximum responses from each cell line by titration curve and growth assay analysis. The responder and non-responder cell lines were each cultured in the same growth conditions. They were then treated with their optimal concentrations of either FGF2 or the FGFR inhibitor, or as a control, with neutral phosphate buffer (PBS), as a vehicle control. Treated cells were incubated for 6, 12 and 24 hours, after which they were lysed in buffer RLT (Qiagen), flash frozen with liquid nitrogen and shipped to us on dry ice. The experiment was split between Merrimack's laboratory facilities in Massachusets and our lab here at CUMC. Cells were treated at Merrimack's facility, then lysed and shipped to us for RNA purification and transcriptome library preparation.

4.3.4 Experimental Protocol

The experimental protocol for this experiment closely mirrored that used in the comparison to the NET drug screen, detailed above, with the following alterations. In this

formulation of the protocol, we used terminal transferase to add adenine nucleotides to the 3' end of the ssDNA strands. These added poly-A sequences were then used as primer annealing sites during second strand synthesis, increasing the efficiency of the reaction by decreasing sequence complexity. This was in response to seemingly poor yields and low complexity in previous libraries.

4.3.5 Results

The sequencing results for this experiment showed comparable gene detection efficiency rates for the two methods. The deeper sequencing run (30M) detected more genes in total than the PLATE-seq (PS) run, which was expected. Encouragingly, the difference in detected genes between these two datasets was small. Roughly 20,000 genes were detected across samples in the TruSeq dataset and between 14,000 and 15,000 genes were detected across samples in the PS dataset. Unfortunately, the inhibitor treatment of the colo699 cells in the TruSeq dataset failed to generate sufficient reads for further analysis, as shown in Fig 4.4. As the TruSeq samples had been processed in our core facility, following standard lab protocols, it was not possible for us to pinpoint what went wrong and we were forced to remove that sample from downstream analysis.

4.4

4.5

4.6

4.4 Using PLATE-Seq to Perform Drug Screening

The identification of drug treatments that are useful in diverse therapeutic settings is a significant driving force in biomedical research [55], [68], [48]. Typical means of measuring the efficacy of a drug for a given clinical application include protein-protein interactions, cell death, mitochondrial respiration and cell growth as well as broader measurements of absorption, distribution, metabolism, excretion and toxicity (ADMET), specifically related the drug or drugs being tested [85]. A wide array of methods are routinely employed to perform these screens, from ligand binding assays [93] to high-throughput proteomics [91].

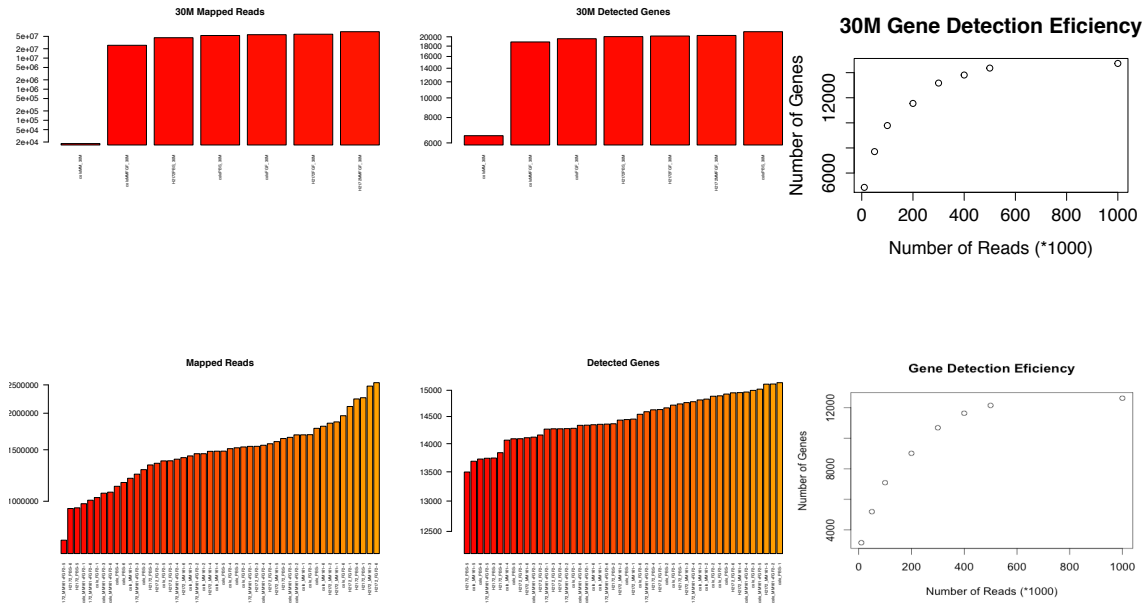

Figure 4.4: Exploratory data analysis for colo699 H2172, at the 24hr time point.

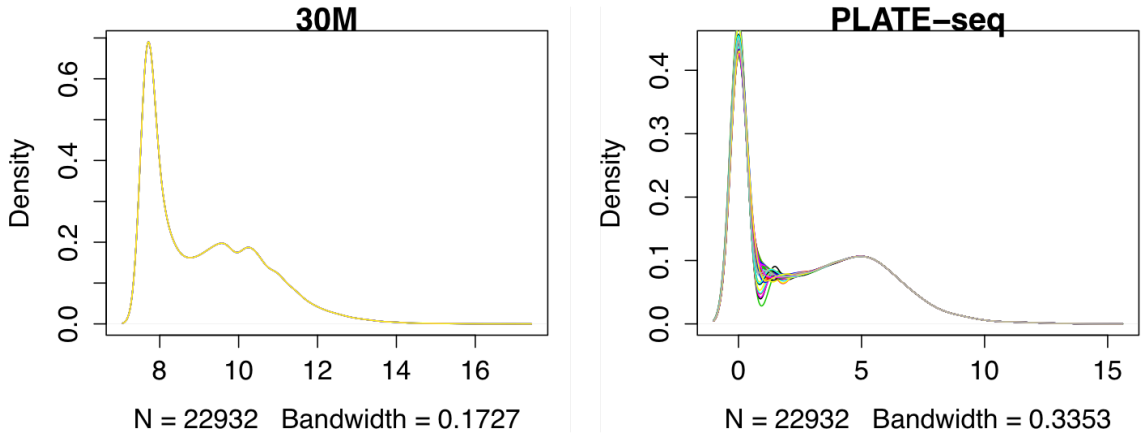

Figure 4.5: Quantile normalized read distributions of PLATE-Seq and TruSeq data.

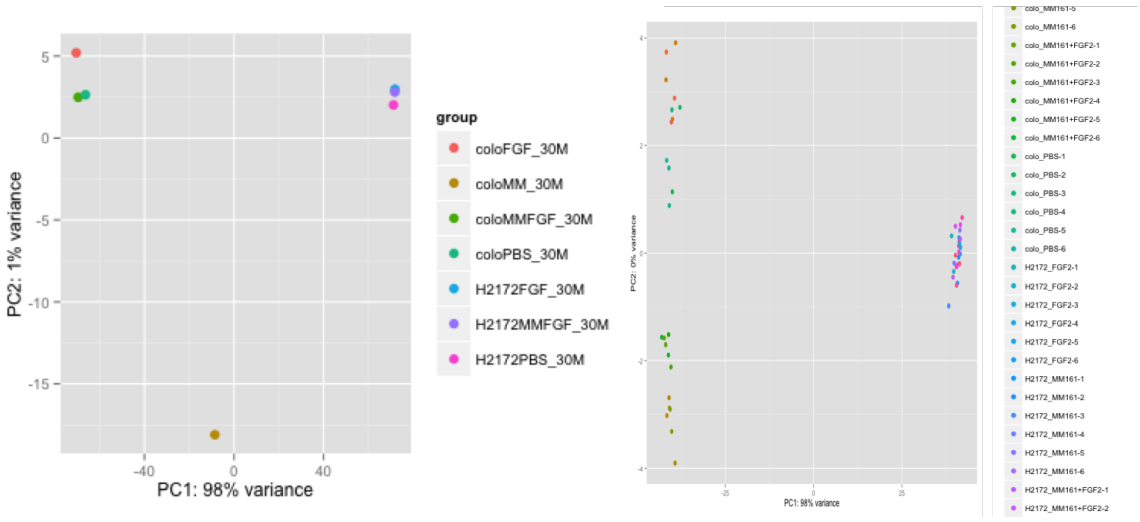

Figure 4.6: Principle component analysis of treatment conditions between responder and non-responder cell lines.

In the end, however, drug screening is an intensive process, essentially composed of a needle-in-a-haystack approach, in which one must sift through thousands of candidate compounds to find only a very few, or in many cases, one single relevant compound (Fig 4.7A). One method that is currently underutilized in small-molecule drug screens and drug discovery is high-throughput transcriptome sequencing (HTS), such as RNA-Seq. Although RNA-Seq is routinely used to profile patterns of genetic changes following perturbations such as drug treatment [99], it has not, to our knowledge, yet been used as the primary readout of a drug screen.

RNA-seq has been used in addition to other biochemical assays as part of an integrated screening pipeline, as in [33], but not as the primary, stand-alone readout of the screen. This is a pity, as RNA-Seq offers many practical advantages (Fig 4.7B). RNA-Seq is a highly scalable technology. There has been considerable interest in making increasingly multiplexed expression libraries to enable researchers to perform increasingly complex and larger-scale sequencing experiments [37], [40], [38]. The primary impediment to making full use of this potential scalability, however, is the price of sequencing, which, while continuing to decline, remains prohibitively expensive for many applications. Performing large-scale

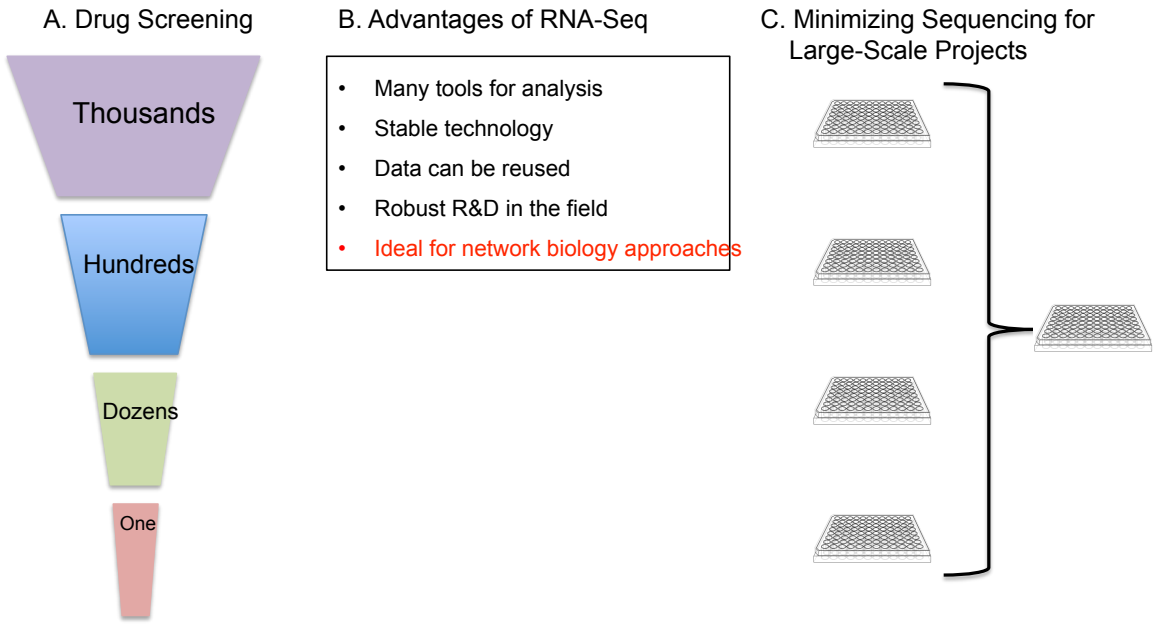

Figure 4.7: A. General drug or other small molecule screening pipeline. From a pool of thousands of candidate molecules, possibly only a single one will prove useful. B. RNA-Seq offers a number of advantages and is the tool of choice for genomic investigations. C. Because of the need for so many different molecules and testing conditions in a large-scale drug screen, there is a strong need to multiplex as many experiments as possible into a single sequencing library.

drug screens is one highly desirable application.

Ultimately, a drug screen encompasses multiple perturbations that have genome-wide effects on cellular networks. The lack of a genome-wide readout is, therefore, a shortcoming in that it represents a lack of potentially very valuable information. This lack is understandable, however, as the cost of sequencing large screens, which must frequently take into account multiple drug concentrations across several time points, is considerable. As we progress in our ability to interrogate cellular networks at a systems level, however, vital techniques such as large-scale drug screens stand to benefit from advances in network biology. To do so, however, they will require genome-scale outputs.

One way to drive down the cost of multiplexing greater numbers of treatments into a drug screen is to lower the depth at which an expression library is sequenced, thereby allowing more samples to be sequenced in any given run (Fig 4.7C). This raises the question of how much sequencing is sufficient for use with network techniques. Here, we show that relatively low-depth sequencing of 500,000 to 2 million (2M) reads per sample is sufficient to acquire informative treatment-specific gene expression profiles. We accomplished this through a highly multiplexed transcriptome library preparation method. As a quality control measure, we compare our method to Illuminaâ^{TM}s TruSeq library preparation protocol.

As a test of the PLATE-Seq method, I compared the gene expression profiles and TF activity profiles that I obtained from treated cells with drugs, using either the PLATE-Seq method or the well-established Illumina TruSeq protocol. The goal was to show that expression libraries obtained via the PLATE-Seq method provided information that was at least equivalent to those obtained by TruSeq, at the level of protein activity network analysis. With the decreased cost of library preparation and concomitant rise in the number of multiplexed samples per library offered by PLATE-Seq, this would demonstrate a practical and cost-effective means for performing high throughput screening at a scale previously unattainable for most laboratories.

4.5 Experimental Set-Up

For this proof-of-principle screen, I selected a panel of seven drugs of known mechanism of action (MoA). These drugs would be administered to BT-20 cells, a basal-like breast carcinoma cell line, at the drugs' IC20 and libraries would be prepared from all samples after 24 hours of exposure. Although PLATE-Seq is designed for screening substantially more treatments in a single experiment, or more conditions per treatment, such as varying concentrations, I was constrained in the number of conditions/treatments that I could test in this experiment by the need to make technical replicates for the TruSeq portion of the samples. The TruSeq protocol calls for a greater per-sample RNA input than does PLATE-Seq. The Sulzberger Genome Core Facility at Columbia University, where the TruSeq fraction was processes, also requires a higher amount of starting material, to hedge against any potential sample loss that may occur during the preparation process. To accumulate the 100ng of required starting material for the TruSeq samples, half the volume of six wells per treatment had to be pooled, making for duplicate biological replicates for each TruSeq treatment condition, versus 12 biological replicates for each condition, as prepared by the PLATE-Seq method. Although 12 biological replicates are well more than sufficient for PLATE-Seq, the abundance of replicates does lend strong statistical power to the resulting analysis.

BT-20 cells were plated in a 96-well plate at an initial density of 8,000 cells per well. They were cultured in an EMEM medium supplemented with 10% FBS and 1% pen/strep. This density was empirically determined to be optimal, based on previous experiments measuring the amount of purified RNA obtained after plating different quantities of cells, as seen in Figure 4.8, grown overnight at 37C and transferred to the automation facility the following morning, allowing for approximately 16 hours of growth to fully recover from the effects of having been split and plated.

Once the plate was in the possession of the automation facility, all following steps until sample pooling were performed fully robotically on the Hamilton MicroLab STAR liquid handling robot. Each row of the plate was administered a separate drug, with final row receiving DMSO, which is the vehicle medium for all drugs in the panel. The protocol was then performed as described in the Methods chapter.

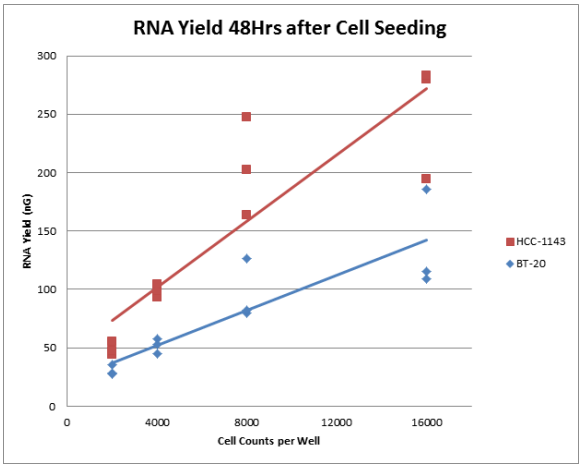

Figure 4.8: RNA purified from cells cultured in 96-well plates

The drugs that were chosen for this experiment were albendazole, aprepitant, bortezomib, gemcitabine, crizotinib, idarubicin and mitoxantrone. Each drug except for idarubicin and mitoxantrone comes from a separate class of drugs and operates in a manner distinct from the others. Conversely, idarubicin and mitoxantrone are both antineoplastics that work by inhibiting DNA synthesis through inhibition of topoisomerases, particularly TOP2A. One of the desired outcomes of this experiment was to see distinct drug profiles at both the gene expression and protein activity levels, from the libraries made using PLATE-Seq. As a form of internal control, I felt that it was also important to see that PLATE-Seq's reduced depth libraries could also accurately capture the similarity that should exist between drugs with very similar MoA.

The specific drugs chosen for this experiment were drawn from a panel of drugs that had been previously used in a much larger drug screen for a separate project taking place in the Califano Lab, called the N-of-One project. This screen was also performed on BT-20 cells, which gave me access to a body of data for use in planning this experiment and will allow results obtained in this experiment to inform future aspects of the much longer running N-of-One project. The drugs selected for this screen were based on the following criteria: 1) they must cluster apart from each other, 2) they must have shown an ability to reverse the gene expression and TF activity signatures, as compared to DMSO

and untreated cells and 3) they must not be the best candidates for further screening in the N-of-One project, so as not to scoop any of the results of that future publication. To this end, the seven drugs selected for this experiment were done so based on a combined method of computational direction and manual curation.

4.6 Results

4.6.1 Data Quality

Two potential sources of significant experimental noise were well-to-well variation in RNA yield and cross talk. To quantify both of these effects, we added ERCC synthetic RNA spike-in controls to alternating wells. This control consists of a mixture of synthetic RNA oligonucleotides of known length, sequence and concentration. Measuring the concentration of each mix component versus the number of recovered reads provided us with a measure of the amount of variation across the plate. Since the spike-ins were loaded into alternating wells, measuring the number of any reads mapping to ERCC sequences in each of the wells provided us with a measure of the amount of cross-talk in the experiment. In the former case, the strong linear relationship between ERCC concentration and normalized counts shows that there was very little variation in RNA yields across the plate (Fig. 4.9). After variance stabilization by DESeq, the fraction of potential ERCC reads mapping to wells in which they were not loaded is negligible.

Although our PLATE-Seq run yielded approximately 2.5% of the number of reads per sample as that of TruSeq (499,513 reads on average in PLATE-Seq vs 19,701,454 reads on average in TruSeq), these reads captured roughly half as many genes as corresponding TruSeq samples (9,875 genes detected, on average, for PLATE-Seq samples vs 16,913 for TruSeq; Fig 4.9A-D), suggesting a more efficient rate of gene detection. Because PLATE-Seq relies on fewer cycles of amplification and a smaller input than TruSeq (12 vs 16, respectively), the number of reads per expressed gene fall closer to the linear range, resulting in a faster gene detection saturation rate. To assess gene detection saturation, we randomly sampled increasing numbers of genes from the raw data and plotted the number of unique gene names that were counted at each iteration (Fig 4.9E). We also performed

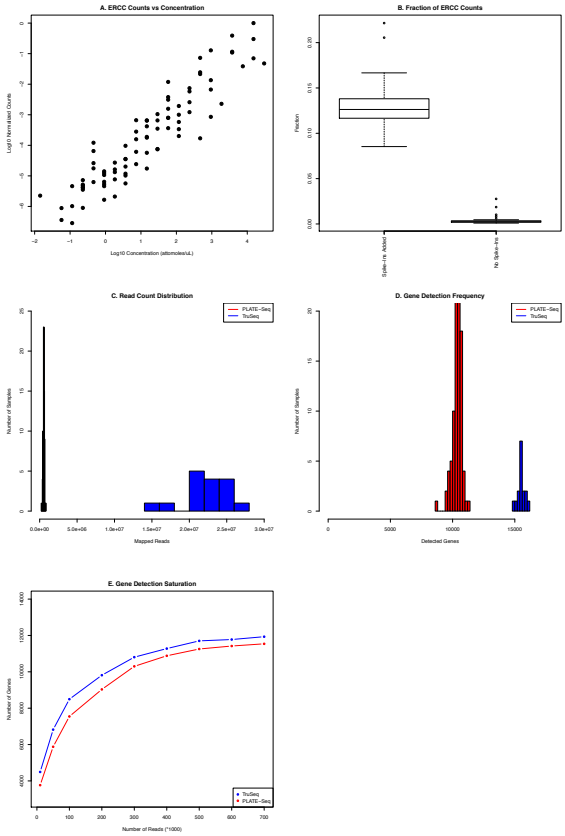

Figure 4.9: ERCC Distribution and Gene Detection Rates

principal component analysis on the GEPs of the TruSeq and PLATE-Seq datasets. In this analysis, a single DMSO sample, shown in Fig 4.10, was seen to be a strong outlier and was therefore removed from the analysis. Following outlier removal, biological replicates of mitoxantrone, bortezomib, crizotinib and idarubicin were seen to cluster together in very tight shot groupings, while albendazole, aprepitant and gemcitabine replicates clustered together with DMSO. Although I had expected albendazole and aprepitant to cluster close to DMSO, as neither of these drugs is known to have profound effects on gene expression networks, I was surprised and mildly disappointed to see the gemcitabine join them, as gemcitabine has a similar MoA to mitoxantrone and idarubicin and is expected to have a correspondingly strong effect on gene expression in treated cells. Nonetheless, a negative result like this one is to be expected from any drug screen of sufficient size. This result merely highlights the natural variability inherent in large molecular screens. Discrepancies between biological replicates in RNA-Seq experiments are not unexpected, owing to a diverse set of factors, ranging from simple pipetting error [60] to naturally occurring stochastic gene expression variation. [64].

Despite such potential for unpredictability, PLATE-Seq compared remarkably well to TruSeq on a number of other QC measures. Differential gene expression was evaluated using DESeq2 [54]. Both datasets displayed a linear decrease in dispersion, or variability in gene expression values, with increased mean normalized counts. Only in the case of the TruSeq library, did the dispersion appear to reach saturation, as seen by the flattening of the curve at the highest range of the mean normalized counts, Fig 4.11. This is expected, given the higher depth of sequencing in that library. Measurements of expression vs fold change and of fold change vs significance, as measured by -log10(p-value), also showed highly compatible results. In both cases, the same genes showed up as being the most significant in both datasets. As a final measure of PLATE-Seq's reproducibility, we noted that gene expression profiles overall compared very positively between the majority of both biological and technical replicates, as seen in Fig 4.12. Taken together, these results gave us confidence in the quality of data obtained through PLATE-Seq.

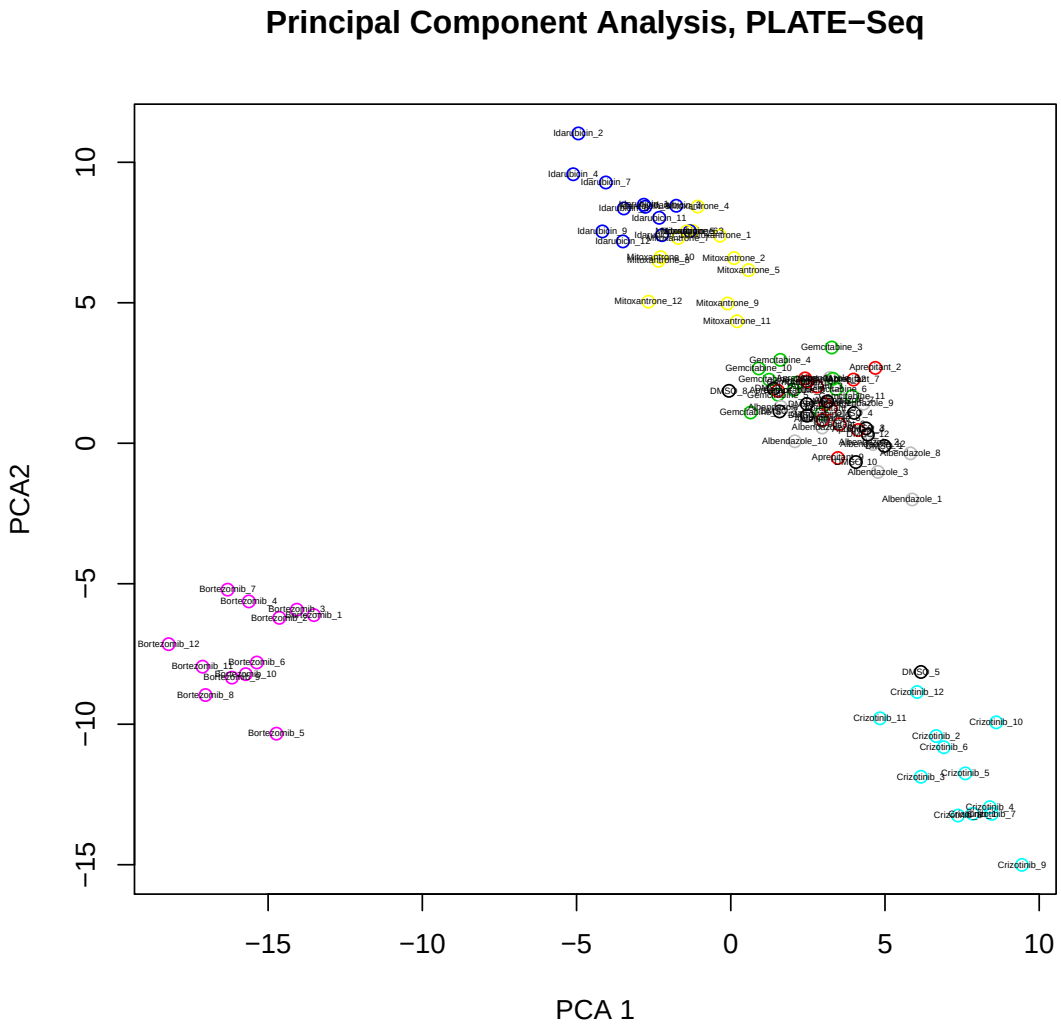

Figure 4.10: Principal component analysis for PLATE-Seq samples, with DMSO replicate outlier

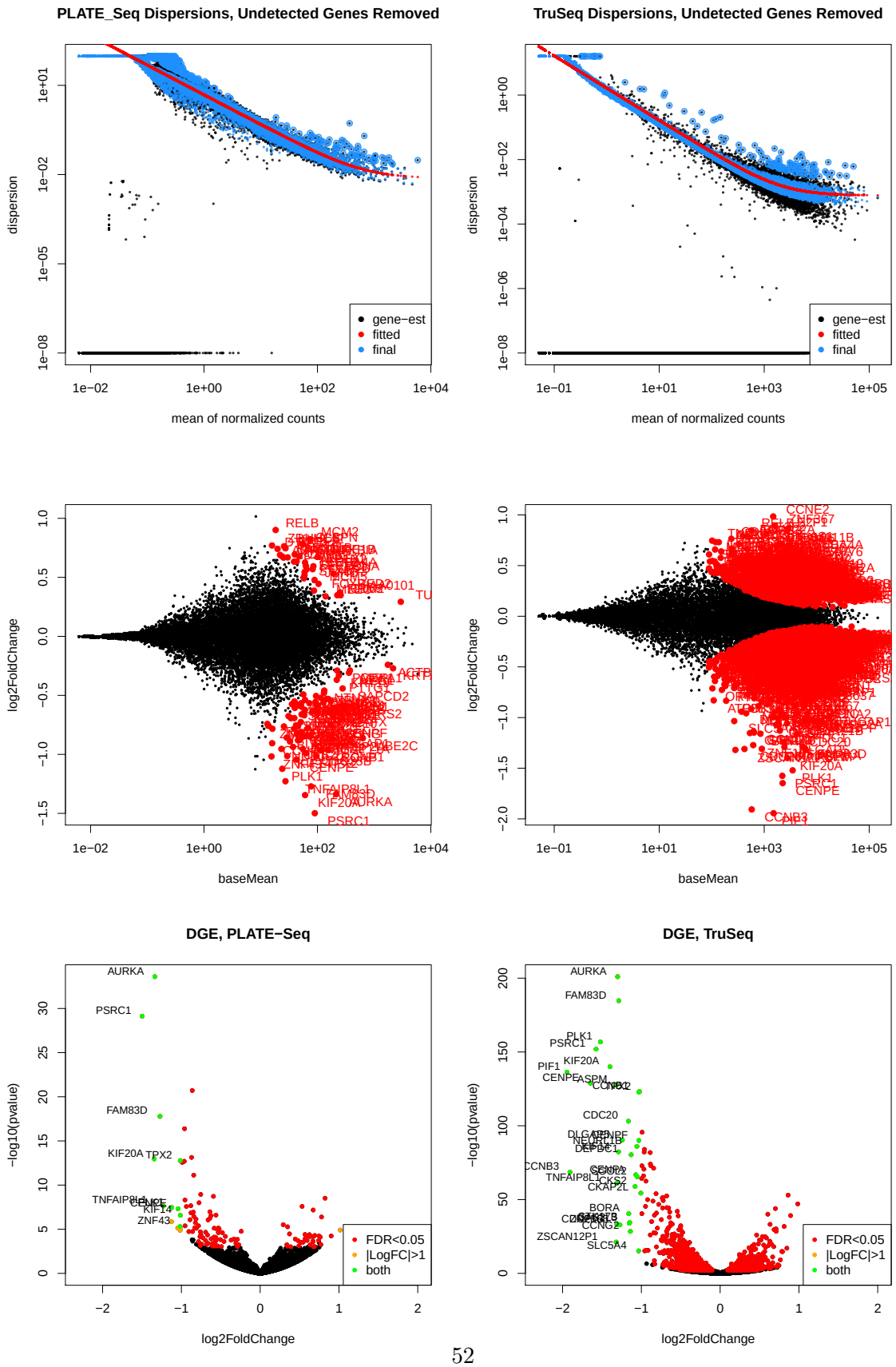

Figure 4.11: Comparison of QC metrics applied to both PLATE-Seq and TruSeq libraries.

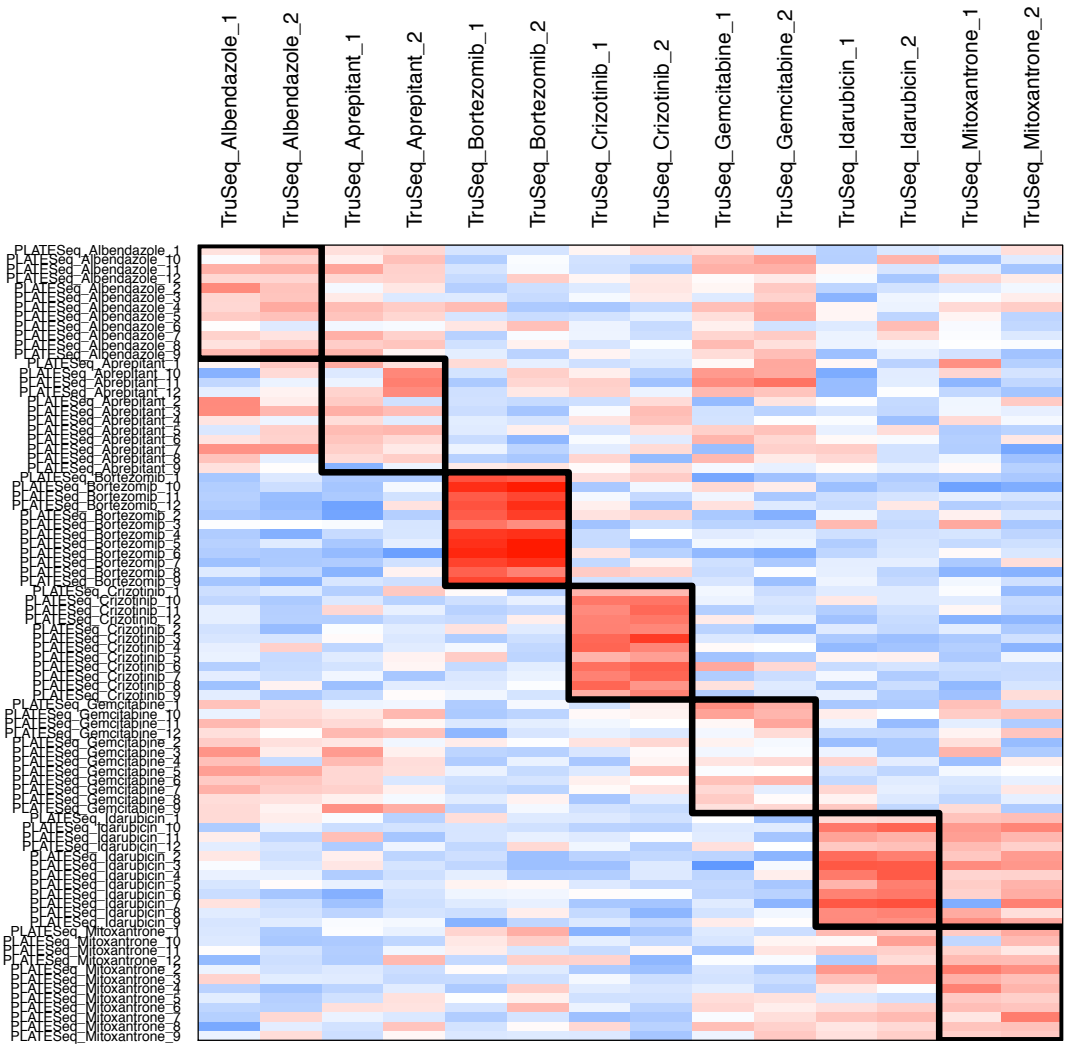

Figure 4.12: Comparison of gene expression signatures across all samples and both platforms.

4.6.2 Large-Scale Drug Screen

Because PLATE-Seq was designed primarily as a tool for large scale screens, its main advantage, that of increasing multiplexing while reducing cost, lies in such a large-scale setting. Our final experiment was therefore to test PLATE-Seq's performance in such a setting and to apply our sequencing results to network biology methods capable of utilizing PLATE-Seq's low read depths. For this, we conducted a larger drug screen and analyzed the results using the VIPER algorithm. Our screen consisted of 69 drug treatments, administered at two concentrations and assayed at two time points. The chosen concentrations were the IC20 for each drug and 1/10 of the IC20. The drugs used in this screen consisted of a broad selection of drug classes, ranging from targeted antibodies, to antivirals to anthelmintic medications directed against eukaryotic parasites (see Table 1 in the appendix). The screen was performed in the same BT-20 cell line as before. The drugs were selected based on cell survival data that had previously been acquired by the Automation Core Facility at CUMC. The procedure for manipulating these plates and constructing libraries from them was the same as that used in the single plate setting.

The procedure used to build this library was identical to that of the smaller-scale library in the previous section. Mapped reads and gene detection rates were also comparable (Fig 4.13). As a further measurement of data quality, we calculated the correlation distributions between all samples and between biological replicates across all plates and plotted them as densities, as seen in Fig 4.14. This shows that wells that were treated with the same drug are more similar to each other than to wells treated with different drugs. We further compared replicate-vs-non-replicate correlations in our PLATE-Seq data to those in other publicly available datasets. For this comparison, we chose drug perturbation expression data from the CMAP and LINCS databases. Although these expression sets were built using different technologies; Affymetrix in the case of CMAP and Nanostring in that of LINCS, all datasets consist of drug perturbation experiments highly similar to our own and contain biological treatment replicates. In theory, any cells that were treated in the same way, with the same perturbagens and under the same conditions, should correlate well to each other. What Fig 4.14 shows it that, compared to the other datasets, replicates assayed using PLATE-Seq are considerably better correlated. This suggests that the data

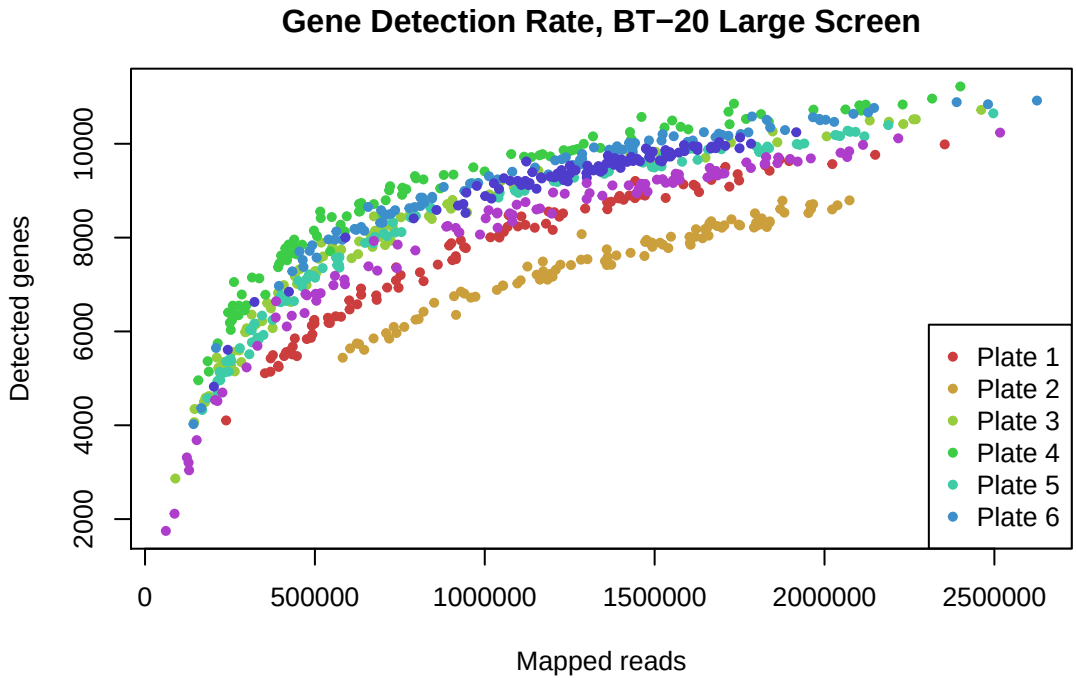

Figure 4.13: Detected genes vs mapped reads for each plate in the large-scale drug screen.

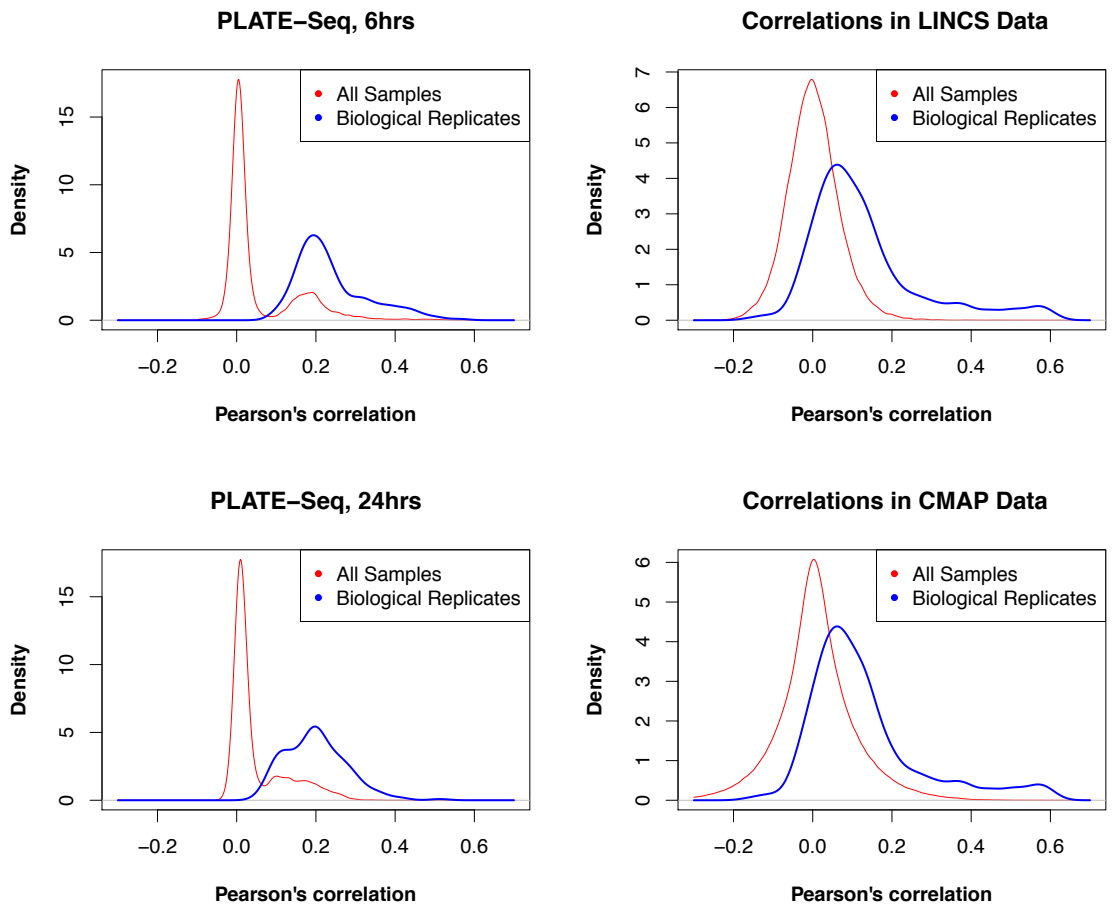

Figure 4.14: Correlation densities between all samples vs between biological replicates.

obtained using PLATE-Seq is both of high quality and is less noisy than other widely-used datasets obtained through prior technologies.

We evaluated sample clustering using the t-Distributed Stochastic Neighbor Embedding (t-SNE) technique. Similar to techniques such as principal component analysis (PCA) or multi-dimensional scaling (MDS), t-SNE is a method for reducing dimensionality. It is particularly well-suited to visualizing scenarios involving high-dimension datasets, such as ours. Analysis by t-SNE showed that samples from the same plate were clearly separate from samples from diverse plates at the level of gene expression, as seen in Fig 4.15. Within each plate-specific grouping, biological replicates of the same concentration tended to clus-

ter together. Less frequently, replicates of different concentrations also clustered together, indicating the effects of each treatment could vary strongly by concentration, in keeping with our current understanding of drug dosing.

To examine the efficacy of VIPER, we selected two drug treatments for closer investigation. For this analysis, we selected vorinostat and temsirolimus, based on their clustering by t-SNE. In the case of both drugs, each pair of replicates of the same concentration clustered tightly together at 24 hours and each treatment was sufficiently far from the other, that we reasoned that we would obtain clearly differentiable results between the two.

4.6.3 Challenges in Applying Virtual Proteomics

The motivation driving the development of PLATE-Seq was to achieve greater experimental throughput using a combination of biochemical and computational methods. Specifically, we believed that lower-depth sequencing could increase the number of experiments sequenced in a single run and that this data would be useful to network techniques such as the virtual proteomics offered by the VIPER algorithm. To this end, I spent a considerable amount of time analyzing our sequencing data with the hope of acquiring verifiable and biologically meaningful results. Doing so proved to be a major challenge and highlights at least one significant obstacle to leveraging the full power of the VIPER algorithm in an investigational setting.

The VIPER algorithm itself consists of a relatively simple and uncontroversial theory: that the differential activity of a TF can be inferred from the differential expression of its transcriptional targets. To put this theory into practice, a sufficiently accurate network of genetic interactions, or interactome, is needed. This is where the problems begin. The accurate identification of TFs and their transcriptional targets constitutes a massive and ongoing undertaking [14]. Different TFs are active under different and specific conditions and identifying these patterns of activity is not a trivial task. Although several techniques have been pioneered to accomplish this task, none comes without significant caveats. These techniques can be split broadly into two classes: biochemical methods and bioinformatic methods.

Despite the sophistication of the methods that have been developed to infer gene

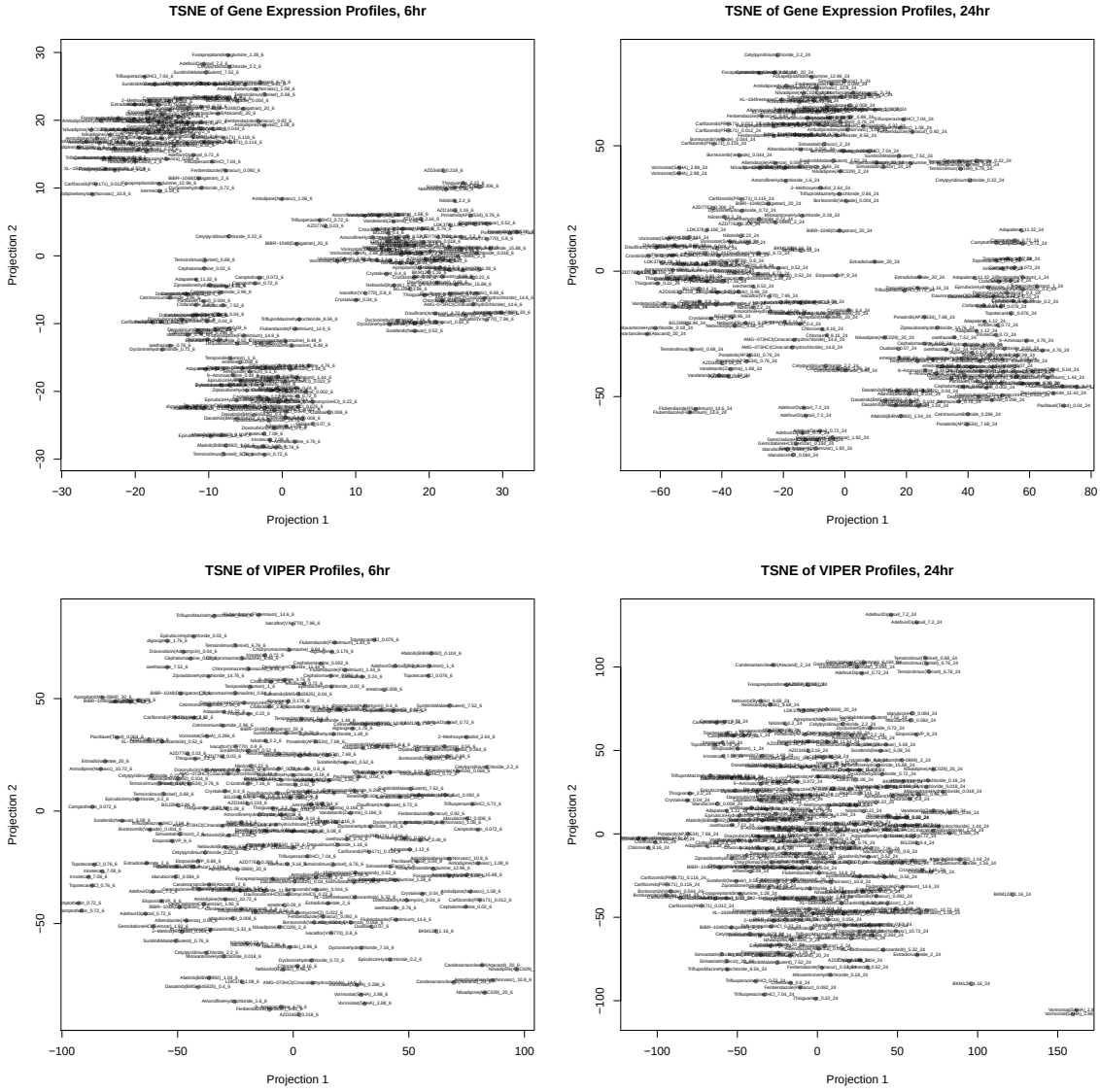

Figure 4.15: Sample grouping based on t-SNE.

regulatory networks, all of those algorithms rely first on the prior identification of which genes constitute putative TFs and which constitute putative targets. The first transcription factors were discovered in focused, single-gene studies [49]. Naturally, we have sought ways to increase the throughput of those experiments in order to build whole gene regulatory networks. Among the biochemical methods, chromatin immunoprecipitation followed by sequencing (ChIP-seq) and ChIP followed by microarray hybridization (ChIP-chip) studies both seek to identify TFs in a high-throughput manner. Positive results in either setting, however, frequently depend on being able to identify other underlying conditions that may lead to DNA binding by the putative TFs [29].

Given the amount of sequence data available and the relatively low overhead costs involved in conducting bioinformatic research, as compared to bench-top biochemistry, a number of bioinformatic methods have been applied to the identification of TF-target interactions. As before, none of these come without caveats. For instance, many algorithms exist to identify and map conserved transcription factor binding sites (TFBS), but sequence conservation is not enough to precisely identify the binding TF ([44], [16], [73]).

My own lab's foundational method, ARACNe, was an attempt to streamline the process of TF-target interaction identification through one of the earliest applications of mutual information theory to the biological sciences. As stated earlier, ARACNe makes the assumption that while not linear, some relationship should exist between the expression of a TF and that of its targets. While this is unlikely to be true on any but the broadest scale, it did enable the most rapid and high-throughput analyses of these relationships to be made at the time. Furthermore, through careful curation of the lists of identified TFs and non-TFs that were used as input and clever experiments, it did serve as a useful method in ibvestigations into the GRNs underpinning some cancer contexts, most notably glioblastoma [10] and in B cells [51]. An unfortunate, but critical negative result of my own studies, which highlights the limits of network inference algorithms, is that our current breast carcinoma networks may not be as reliable we need them to be.

A considerable amount of time was spent analyzing VIPER results that I obtained using a breast carcinoma interactome that was composed of three classes of regulators: TFs, co-TFs and signaling molecules. The TF component was obtained as described pre-

viously. What I only discovered later, was that the co-TF network was inferred not from direct experimental data, but by GO terms [2] that were deemed likely to be associated with transcription factors. The specific GO terms used to generate the co-TF list were GO:0003712 (transcriptional coregulation) and GO:0005634 (nuclear localization). While these terms may represent many co-transcription factors, it is unlikely that all genes represented by those terms will actually be co-TFs, thus adding noise to an already noisy system. The signaling proteins and their targets composed an even more abstracted set of potential regulatory interactions than that of the co-TFs. This network was inferred by running ARACNe on RNA-Seq data and using a list of known signaling molecules rather than one composed of transcription factors. Although the ARACNe algorithm will treat these inputs the same and look for statistically significant relationships between the list of regulators and the other genes in the RNA-Seq raw counts data, this resulting statistical relationships are unlikely to have any biological relevance, since the methods itself rests on an untrue assumption; that the expression of signaling molecules (predominantly phosphatases and kinases) in any way tracks that of differential gene expression. Signaling molecules operate on much smaller timescales than those relevant for gene expression. Because phosphorylation reactions are measured on the order of milliseconds to seconds [86], while transcription takes minutes to occur [58], RNA-Seq data that is not part of a time course trial stands little to no chance of catching measurements on the relevant timescales.

As an example of the perils of building a network in this way, we found that VIPER failed to detect any differential activity whatsoever of mTOR, the extremely well-characterized target of the rapamycin analogue temsirolimus [12], [41], [76], [95], as seen in the bottom of Fig 4.16. A further challenge came in trying to reconcile the other regulators inferred as most differentially active with the gene products known to interact with that drug or its targets. At best, we could only note that some of the inferred regulators operated within the same pathways known to be perturbed by the treatments. Even in this case, however, most of these pathways were related to cell division and cell death, both of which are known to be perturbed in all cases where a drug treatment stresses cells. Because of this, we were unable to unambiguously separate the action of any specific drug from that of any drug whatsoever.

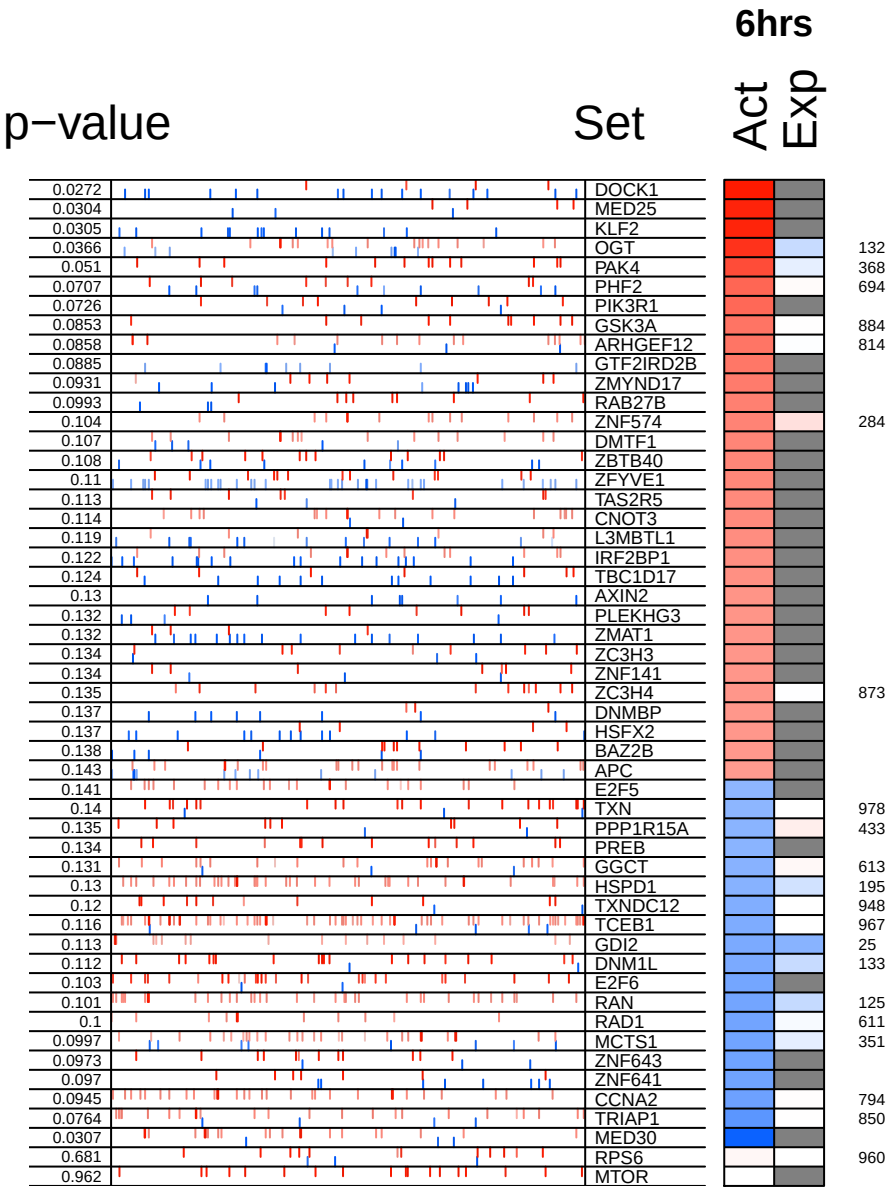

Figure 4.16: The results of master regulator analysis for BT-20 cells treated with temsirolimus, a rapamycin analogue.

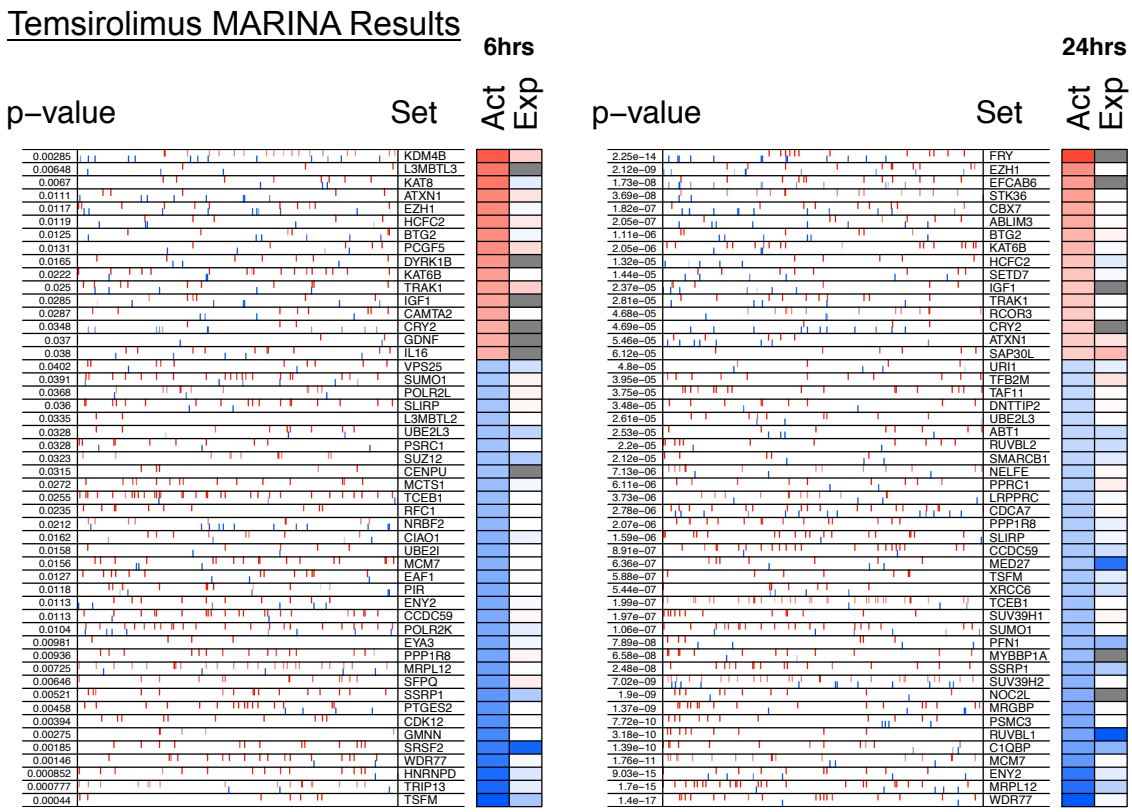

Figure 4.17: The results of master regulator analysis for BT-20 cells treated with temsirolimus, using an interactome built using only TFs and co-TFs.

To improve the results of VIPER analysis, I first removed the signaling genes from the breast carcinoma interactome. For this analysis, I chose to focus on two treatments, temsirolimus and vorinostat, as replicates from these two appeared to cluster tightly together, while the two treatments separated well from each other, as seen in Fig 4.15. The results from this analysis were not significantly easier to explain. In the case of temsirolimus, the only potentially useful finding was that a slight increase in activity was measured for the gene p27, which would be expected from mTOR inhibition (Fig 4.17). Even so, the inferred activity for p27 was not strong and worse, no other differential activity was found in other known targets of the mTOR pathway.

Vorinostat is an inhibitor of types I and II histone deacetylases (HDACs) [71]. Al-

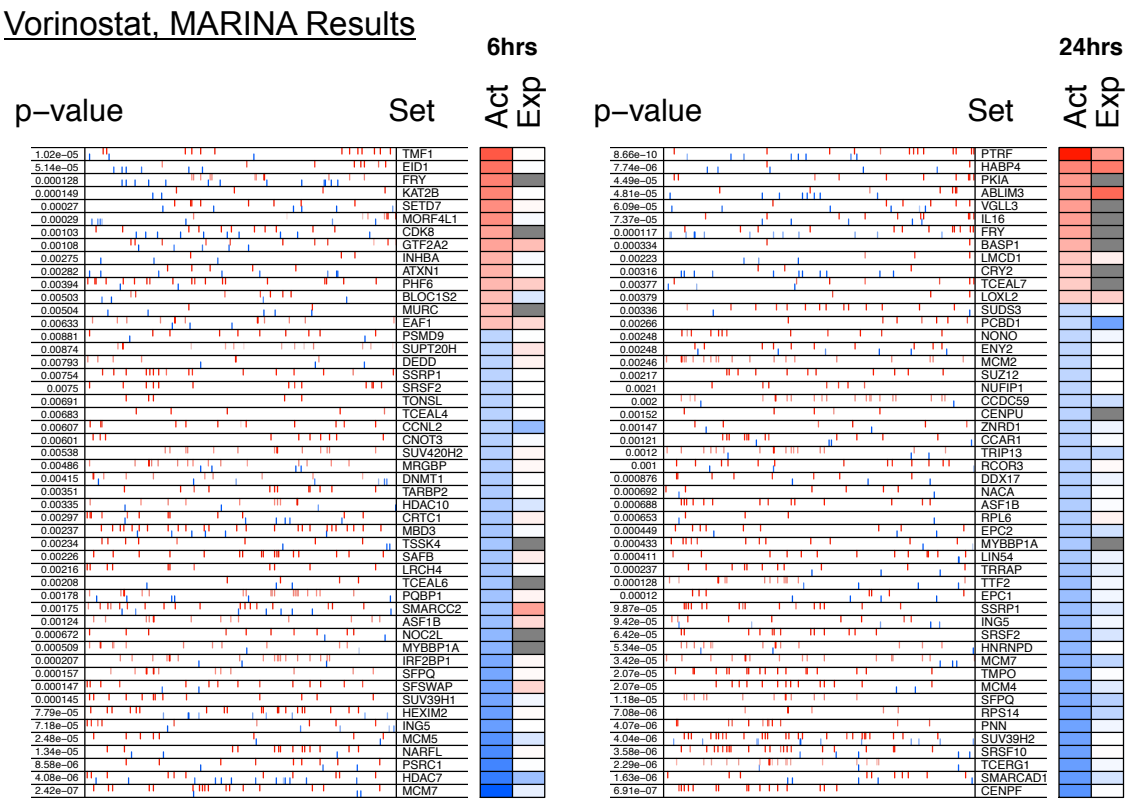

Figure 4.18: The results of master regulator analysis for BT-20 cells treated with vorinostat, using an interactome built using only TFs and co-TFs.

though in the analysis of this treatment, using the TF and co-TF combined interactome, we inferred the differential down-regulation of two HDACs, HDAC7 and HDAC 10 (Fig 4.18), we could find no evidence that these were known targets of vorinostat. Known HDAC targets were not found to be differentially regulated in this dataset. Perhaps interestingly, we noticed that the down-regulation of HDAC7 and HDAC10 peakeded 6 hours, which is in keeping with the pharmacokinetics of vorinostat, in which the mean serum concentration has been observed to peak at roughly 6 hours [39]. In the absence of more evidence, however, we felt that these data made a very unconvincing argument for the efficacy of this analysis as a screen for drug effect.

Finally, I trimmed the interactome to only TFs, as the ARACNe-inferred TF-target interactions are arguably the most likely of the categories to be biologically true and rele-

vant. Although TF expression does not linearly correlate with target expression, if they are related at all, this network presents the least abstracted set of possible regulatory interactions of the overall breast carcinoma interactome. The results of this analysis were slightly more encouraging than those of the last two, although were still somewhat shy of being truly convincing. Figure 4.19 shows that we inferred the differential activity of one known, one suspected and one possible regulatory targets of mTOR inhibition. The known target of mTOR inhibition is HMGA1, which we found to be differentially down-regulated at 24 hours post-treatment. mTOR is known to upregulate the expression of HMGA1 [6]. It remains unclear whether we should actually expect to infer a reduction in the transcriptional activity of HMGA1 24 hours after mTOR inhibition, although this seems like a reasonable hypothesis. FOXM1 and STAT6 were found to be down- and up-regulated, respectively. Increased transcriptional activity of FOXM1 is a known downstream effect of mTOR phosphorylation by AKT [52], [70], although to our knowledge, the effect of mTOR inhibition on FOXM1 has not been directly investigated, relegating this result to being logically plausible, but unconfirmed. A possible regulatory interaction between mTOR and STAT6 has been suggested in the past [75], but to our knowledge, never confirmed. Finally, PA2G4/EBP1 was inferred to be strongly down-regulated by 24 hours. The ambiguity of these master regulator analysis results, coupled to the clear clustering pattern observed in the t-SNE plot, suggest that the quality of our available network represents a significant impediment to the application of network methods such as VIPER.

Despite these obstacles, we have successfully developed an efficient and scalable protocol that is cheap enough to allow RNA-Seq to be used as a readout for HTS screens, which enables us to identify drugs that affect gene expression.

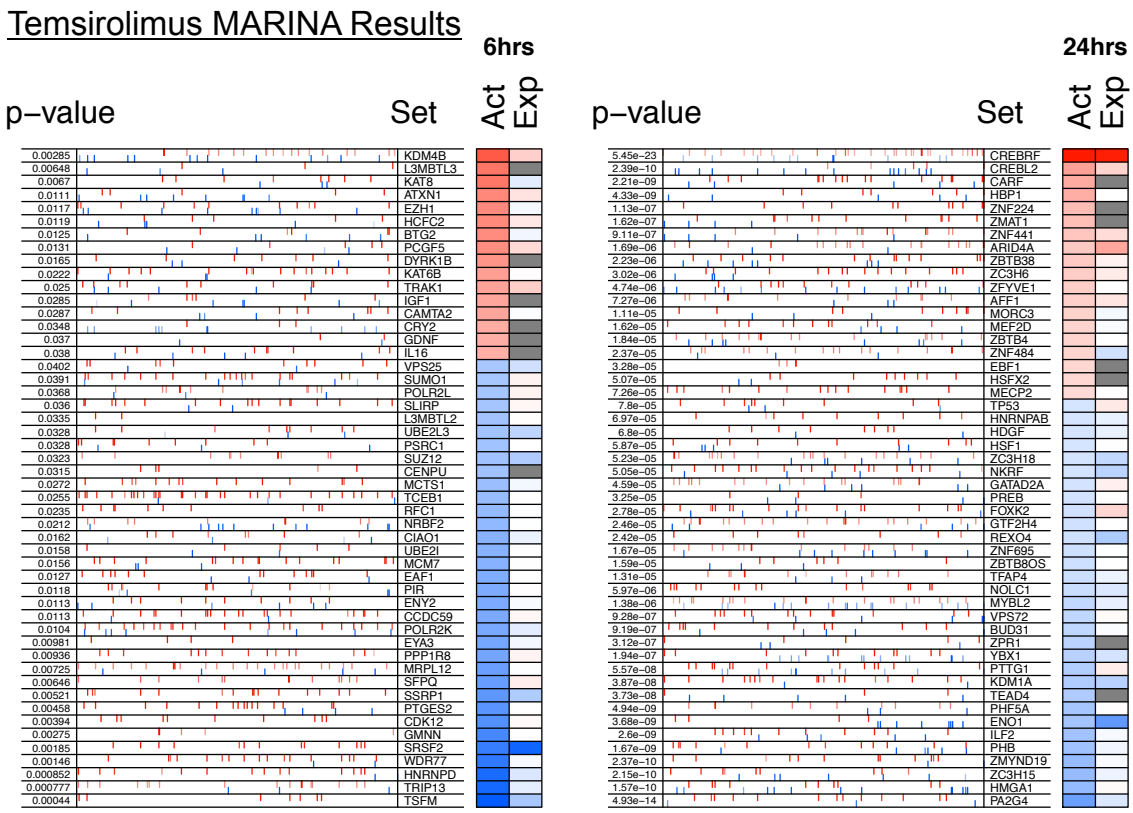

Figure 4.19: The results of master regulator analysis for BT-20 cells treated with temsirolimus, using an interactome built using only TFs.

Chapter 5

Conclusion and Discussion

PLATE-Seq is a highly efficient expression library preparation method that can be readily adapted to a number of experimental pipelines.

5.1 Clinical Applications

As method for performing drug screening in an expedient and low-cost manner, PLATE-Seq is an obvious target for clinical applications. In particular, PLATE-Seq and its related suite of analytic tools are well-suited to assisting in the investigation of two clinical objectives:drug repositioning and the search for personalized drug therapy regimens.

5.1.1 Drug Repositioning

The past decades have seen a significant decline in the rate of new drug discovery [62]. Discovering a novel therapeutic compound and successfully shepherding it through the trial and approval process requires significant investments of time and money and suffers from a very low rate of success [22]. One strategy that has been gaining traction in recent years consists of finding novel uses for drugs that are already on the market, a process variably known as drug repurposing or drug repositioning [13] (Fig 5.2).

The primary advantages to drug repositioning are reduced costs and development times. It is estimated to take roughly 17 years and $2.6 billion dollars to bring a new drug to market in the United States [59]. With the advent of drug repositioning, the discovery

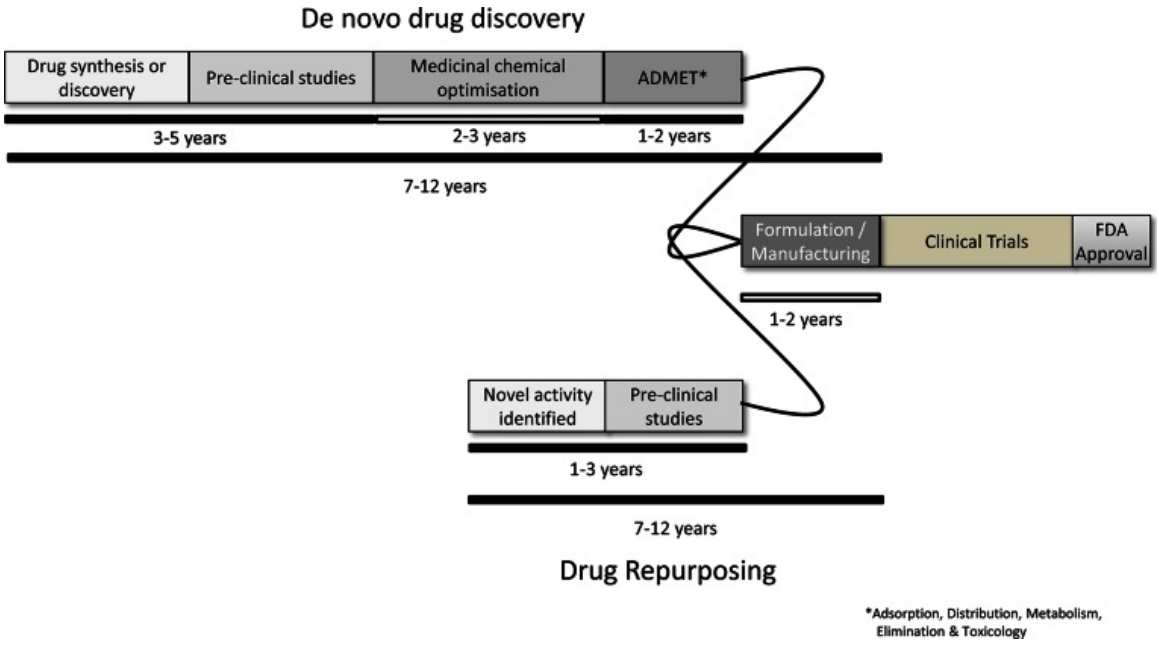

Figure 5.1: Timescales involved in drug discovery vs repositioning, McCabe, et al., 2015

of novel applications for combinations of drugs takes on average only 3 - 12 years (Fig 5.1). Repositioning also offers the distinct advantage of beginning a study with compounds that are already proven to be safe and for which a wealth of information concerning pharmacokinetics, toxicities, bioavailability and dosing is already available, essentially bypassing Phase I of the clinical testing process.

Despite the cost savings involved in beginning a drug trial with drugs that are already proven safe, testing all possible combinations of a sufficiently large set of drugs remains a significant financial consideration. Testing pair-wise combinations of the roughly 2,000 FDA-approved drugs [46] in one cell line and at a single concentration and time point would result in nearly 2 million combinations. Realistically, each drug pair would have to be tested in multiple conditions, vastly increasing the cost of the investigation. Although not all drugs might need to to tested in combination in a given investigation, even a more reasonable number, such as 100 drugs represents a large endeavor. This is the ideal setting for a technology such as PLATE-Seq, which offers a high-throughput genome-based means for interrogating drug effects at reduced cost and increased multiplicity. No single drug

COMPOUND	COMMON DISEASE	ORPHAN DISEASE
Azathioprine	Rheumatoid arthritis	Renal transplant
Bleomycin	Various cancers	Pleural effusion
Colchicine	Gout	Mediterranean fever
Cycloserine	Urinary tract infection	Tuberculosis
Cyclosporine	Rheumatoid arthritis Psoriasis	Transplant rejection
Eflornithine	Unwanted facial hair	Sleeping sickness
Everolimus	Renal cancer	Renal transplant
Histrelin	Prostate cancer	Precocious puberty
Infliximab	Ulcerative colitis Rheumatoid arthritis Psoriasis	Crohn's disease
Interferon alfa	Hepatitis B and C	Various cancers
Rituximab	Rheumatoid arthritis	Various cancers

SOURCE: Food & Drug Administration

Figure 5.2: Drugs that have been repositioned for use beyond their original specifications.

repositioning investigation need be comprehensively large on its own as long as more labs have the means to pursue these investigations. Another benefit of RNA-Seq data is that it exists in an easily shared digital form. The more that labs are able to pursue drug repositioning studies, the more information will be available to all, allowing a robust use of what promises to be a rapidly expanding knowledge base. PLATE-Seq promises to serve as a key component of the already growing drug repositioning effort, as well as that of the related topic of drug synergy.

5.1.2 Drug Synergy

One of the most desired outcomes of testing combinations of drugs is the identification of those combinations, in which the joint action of the combined drugs is greater than the sum of their constituent parts. Broadly, there are three outcomes to any drug combination. First, the drugs in question may perform in an additive manner, wherein their combined effects amount to the sum of their individual effects. Second, they may interact antagonistically, reducing the effects of each drug. Finally, they may act superadditively. It is this last effect that is considered most desirable in any drug combination. Combination drug therapy has long been used to treat a number of ailments, most notably cancer, where single-drug chemotherapy regimens have, in fact, become quite rare.

The goal of combination therapy is to treat a single disease with multiple therapeutic

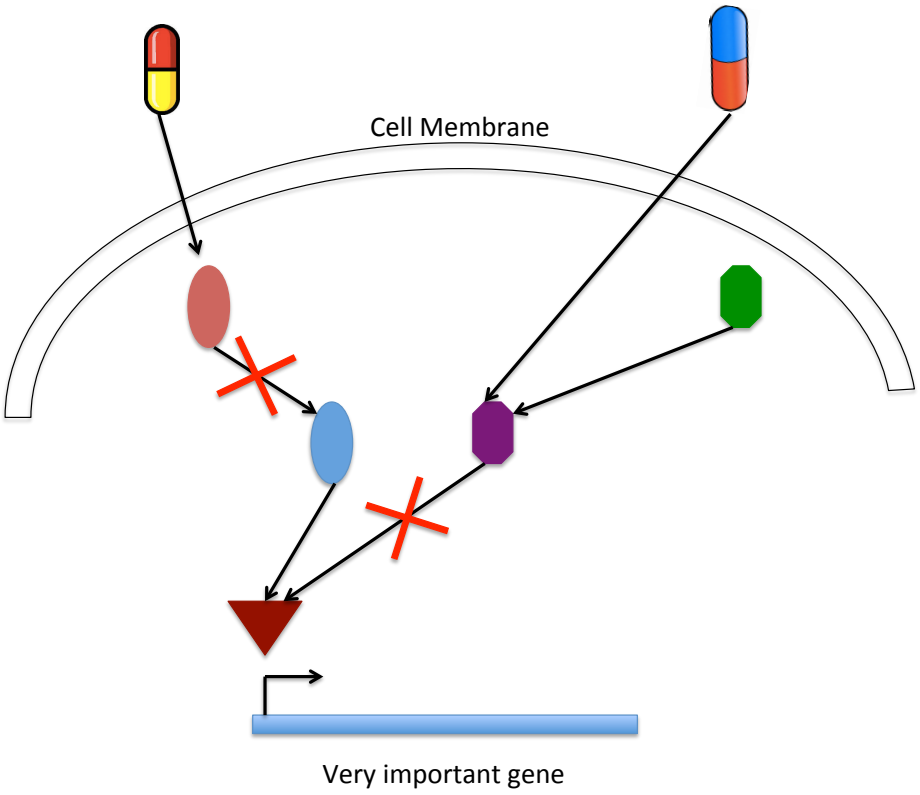

Figure 5.3: Drug synergy occurs when drugs affect different pathways that lead to the same cellular outcome.

agents. A significant reason that synergy is a desirable effect is because a synergistic effect is one in which drugs act on different targets that lead to the same end result, thereby overcoming a cellular network's natural robustness to perturbation (Fig 5.3). Regulating another compound's absorption and distribution, modulating the cell's growth properties, inhibiting compound degradation and inhibiting pathways that induce resistance or reducing the other compound's toxicity are some of the ways, in which one compound might sensitize a cell to another compound, leading to synergistic effects. [5]

Establishing synergistic effects is not a trivial task. Not only are there millions of potential drug combinations [53], but the dose at which a given drug proves both safe and effective in a single-drug setting may differ from its most effective dose in combination with

another drug [84], adding a considerable layer of complexity to the matter. This requires testing each drug in a candidate synergy pair at multiple concentrations, after having first tested numerous combinations to identify this candidate pair. Combinatorial drug testing on this scale remains financially and logistically prohibitive.

The integration of genomic and proteomic information can greatly facilitate the search for synergistic drug combinations. Every drug will have defined effects on cellular networks, both at the level of gene expression and of protein activity. Given sufficiently accurate networks, describing the characteristic alterations that are specific to each drug at each dose will allow predictions to be made, concerning which drugs to test in combination, leading to a more targeted approach. Nonetheless, using commercial transcriptome library preparation kits and sequencing to high depth still represents a cost that most labs simply cannot afford. The PLATE-Seq + VIPER pipeline of transcriptome library prep and virtual proteomics can significantly streamline this process. Using TF activity, rather than full gene expression profiles to characterize drugs' effects first allows for many more drug effect profiles to be acquired through expression library sequencing. Predictions of potential synergistic pairs can be based on these profiles and each pair tested across a range of concentrations and time points.

5.1.3 Personalized Medicine

One of the strongest ideas to pass through the field of biomedical research is the concept of personalized medicine, wherein therapies are tailored to an individual based on their unique genetic profile [31]. The idea of personalized treatments has taken particular hold in the field of cancer research, where the current understanding of cancer is that each cancer is specific to the afflicted individual. To date, genomic information only constitutes one part of a highly integrated set of data used to compile a profile of a person's health. These data include health risk assessment, family health history, and clinical decision support for complex risk and predictive information. Personal genomic data remains a small part of a patient's complete medical profile for two reasons. The first is that it is easily the hardest information to collect and the second is that we are still learning how to interpret personal genomic data in actionable ways [72], [3].

A significant challenge in interpreting personal genomic results stems from small sample sizes. This could be improved with access to cheaper, faster genomic diagnostic tools, of which PLATE-Seq promises to be a part. Although PLATE-Seq can certainly be adapted to a clinical setting, it is important to take stock of its limits in this regard. First, because the economical gains of this technique stem from sequencing at a very low depth, it cannot be used to look for mutations or SNPs. However, preliminary data suggest that little sequencing depth is needed to classify patients into different disease subtypes, such as in the case of breast cancer [74], [97]. Because disease subtypes bear diagnostic value [83], [36], as those subtypes become more complete and as our understanding of the molecular profiles of disease subtypes grow, techniques such as PLATE-Seq are well-positioned for vital future clinical uses.

One obstacle to overcome in developing the PLATE-Seq method into a clinical tool consists of adapting the tissue culture methods involved in library preparation for use with fresh patient-derived tumor biopsies. It is well-known that cell lines are imperfect models of the conditions that regularly afflict patients. It would be considerably better to be able to test biological responses to treatment regimens directly on excised patient tissue. Although this would not address all questions of drug delivery and penetrance, it is difficult to imagine anything that could serve as a better model for patient cell response. Several technical challenges would have to be overcome to achieve this goal. A protocol to extract and transfer the tissue would need to be established, followed by one to administer treatments and disrupt the tissue to extract high-quality mRNA from it, which is not always a trivial task in the context of primary tissue samples.

5.2 Dissecting the Physiographical Landscape of Gene Regulatory Interactions

With effective mRNA extraction procedures, another interesting application of PLATE-Seq would be the three dimensional mapping gene regulatory landscapes of complex tissues, from solid tumors to healthy but heterogeneous tissues such as brain slices. Intratumor heterogeneity is a well-known phenomenon [101], thought to occur predomi-

nantly through the expansion of different clonal populations. Another driver of intratumor heterogeneity is the tumor environment itself [65]. The interior of a tumor is a highly anoxic place compared to its edges and the diverse challenges presented to the cells in either environment will elicit diverse network effects, likely resulting in distinct drug responses. It would be highly informative to be able to map the changing genetic regulatory landscape of these tumors as a function of physiographic location. Such an approach could be extended to complex tissues other than tumors, such as liver and brain tissue. Besides the practical knowledge that this would bring concerning drug sensitivity and response, this would also shed light on fundamental questions surrounding the complex interplay between environmental conditions and cellular development.

5.3 Pooled Expression Libraries of Pooled shRNA Libraries

Similar to chemical perturbations, RNA interference, or RNAi, is a powerful and well-established way of interrogating gene function [80]. One advantage that RNAi has over small molecule chemicals is that RNAi perturbations are highly targeted to specific nucleotide sequences, whereas this is not the norm with small synthetic chemicals. Recent advances in the development of pooled shRNA libraries have enabled the parallel analysis of populations of cells, wherein different genes are suppressed in different cells [79], allowing for a much more fine-grained analysis of gene function than has previously been possible. In the da Silva study, the authors sought to identify genes essential for synthetically lethal genetic interactions through shRNAs that were selectively depleted from populations. To this end, they tested pools containing 6,000, 10,000 and 20,000 shRNA constructs at different scales and at varying levels of population complexity. They infected cells with their pooled libraries at sufficiently low multiplicities of infection (MoI), with the goal of infecting each cell with, on average, a single shRNA-bearing virus. With each shRNA barcoded for deconvolution later in the process, gene expression profiling was performed using microarrays and the data was analyzed using the EDGE package of microarray analysis tools [50]. Genes identified as candidates from this analysis were further investigated by a second screen, in which candidate gene-specific shRNA sequences were coupled to a GFP

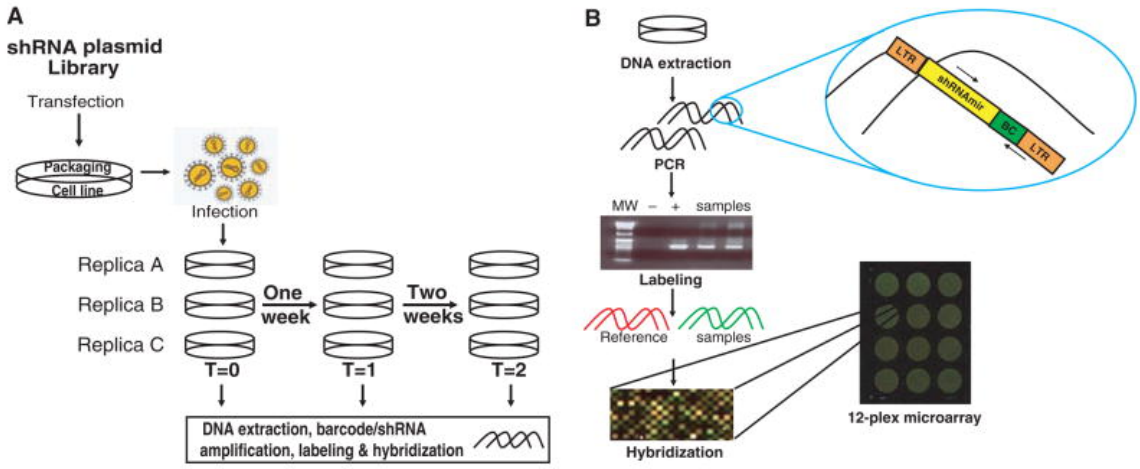

Figure 5.4: Pooled shRNA screening process. Da Silva, et al., 2008

reporter, to link shRNA expression with target gene knockdown efficiency (Figure 5.4).

A key strength of this study lay in demonstrating an actionable method for performing simultaneous genome-wide knockdown studies. Despite the high degree of multiplexing in library preparation, however, the cost and labor of this study were increased by the authors' reliance on microarrays for differential gene expression analysis and on GFP-coupled reporter vectors. A number of advantages would be gained by coupling this shRNA pooling scheme to the PLATE-Seq pooled expression library method.

The chief advantage of the pooling of these pooled library strategies lies in multiplicity. Twenty thousand bar-coded shRNAs distributed among 4,000 to 8,000 cells per well in a 96-well plate could easily translate into between 384,000 and 768,000 single-gene knockdown expression profiles. This scale of gene silencing offers a means to rapidly generate the most fine-grained GRNs to date. To paraphrase Tom hanks in A League of Their Own, if it was easy, everyone would already be doing it. The greatest challenge in pursuing this 'pool of pools' expression profiling strategy lies in deconvoluting not only which bar-coded shRNA comes from which well, but from which individual cell, as their would be a heterogeneous mix of cells in any population, each infected with a distinct shRNA construct.

Although the challenges of properly constructing and deconvoluting such a library are not trivial, the technology exists to act upon them. History has shown repeatedly that

where desire is matched by means, ingenuity does not lag far behind. Arguably the greatest challenge in systems biology is that of constructing networks that accurately represent the biological systems that they were designed to model. Genome-wide combinatorial gene silencing would be a significant step forward in constructing such networks.

One disadvantage of using RNAi concerns the occurrence of off-target effects. The efficiency for any given siRNA construct is somewhat unpredictable and therefore construct testing is usually required to accurately interpret results. In a pooled setting, some ambiguity in target specificity is inevitable, as no reasonable method exists to test siRNA constructs at the magnitude required for high-throughput screening. Pooled CRISPR [78], [17] libraries, while being confronted with the same design challenges as in the case of shRNA libraries, would at least prove to be an improvement over RNAi in terms of target specificity .

Systems biology is poised to enter the realm of truly big data, promising to make powerful use of greater and greater amounts of data, while simultaneously making the results ever more relevant on a very personal level. PLATE-Seq and related technologies are going to play an important role in the development of this field, which is still very much in its infancy. We are, in essence, still learning to speak the language of the cell as a complete system and the technology that we pursue will bring us ever closer to fully understanding that language.

Bibliography

[1] Simon Anders and Wolfgang Huber. "Differential expression analysis for sequence count data". In: *Genome biol* 11.10 (2010), R106.

[2] Michael Ashburner et al. "Gene Ontology: tool for the unification of biology". In: *Nature genetics* 25.1 (2000), pp. 25–29.

[3] Euan A Ashley et al. "Clinical assessment incorporating a personal genome". In: *The Lancet* 375.9725 (2010), pp. 1525–1535.

[4] Alvaro Aytes et al. "Cross-species regulatory network analysis identifies a synergistic interaction between FOXM1 and CENPF that drives prostate cancer malignancy". In: *Cancer cell* 25.5 (2014), pp. 638–651.

[5] Mukesh Bansal et al. "A community computational challenge to predict the activity of pairs of compounds". In: *Nature biotechnology* 32.12 (2014), pp. 1213–1222.

[6] Jamie G Bates et al. "Extensive gene-specific translational reprogramming in a model of B cell differentiation and Abl-dependent transformation". In: *PloS one* 7.5 (2012), e37108.

[7] Matthew J Beal et al. "A Bayesian approach to reconstructing genetic regulatory networks with hidden factors". In: *Bioinformatics* 21.3 (2005), pp. 349–356.

[8] Rotem Ben-Hamo and Sol Efroni. "Gene expression and network-based analysis reveals a novel role for hsa-miR-9 and drug control over the p38 network in glioblastoma multiforme progression". In: *Genome Med* 3.11 (2011), p. 77.

[9] Atul J Butte and Isaac S Kohane. "Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements". In: *Pac Symp Biocomput*. Vol. 5. Citeseer. 2000, pp. 418–429.

[10] James C Chen et al. "Identification of causal genetic drivers of human disease through systems-level analysis of regulatory networks". In: *Cell* 159.2 (2014), pp. 402–414.

[11] Li Chen et al. "Identifying cancer biomarkers by network-constrained support vector machines". In: *BMC systems biology* 5.1 (2011), p. 1.

[12] XF Cheng et al. "Expression of mTOR and its inhibitory effect on cell proliferation and apoptosis of breast cancer cells." In: *Journal of biological regulators and homeostatic agents* 29.4 (2014), pp. 869–873.

[13] Curtis R Chong and David J Sullivan. "New uses for old drugs". In: *Nature* 448.7154 (2007), pp. 645–646.

[14] Gordon Chua et al. "Identifying transcription factor functions and targets by phenotypic activation". In: *Proceedings of the National Academy of Sciences* 103.32 (2006), pp. 12045–12050.

[15] Han-Yu Chuang et al. "Network-based classification of breast cancer metastasis". In: *Molecular systems biology* 3.1 (2007).

[16] Paul Cliften et al. "Finding functional features in Saccharomyces genomes by phylogenetic footprinting". In: *science* 301.5629 (2003), pp. 71–76.

[17] Le Cong and Feng Zhang. "Genome engineering using CRISPR-Cas9 system". In: *Chromosomal Mutagenesis* (2015), pp. 197–217.

[18] Thomas M Cover and Joy A Thomas. "Elements of information theory 2nd edition". In: (2006).

[19] James E Darnell. "Transcription factors as targets for cancer therapy". In: *Nature Reviews Cancer* 2.10 (2002), pp. 740–749.

[20] Matthias Dehmer, Laurin AJ Mueller, and Frank Emmert-Streib. "Quantitative network measures as biomarkers for classifying prostate cancer disease states: a systems approach to diagnostic biomarkers". In: *PloS one* 8.11 (2013), e77602.

[21] Santiago Diaz-Moralli et al. “Targeting cell cycle regulation in cancer therapy”. In: *Pharmacology & therapeutics* 138.2 (2013), pp. 255–271.

[22] Joseph A DiMasi, Ronald W Hansen, and Henry G Grabowski. “The price of innovation: new estimates of drug development costs”. In: *Journal of health economics* 22.2 (2003), pp. 151–185.

[23] Alexander Dobin et al. “STAR: ultrafast universal RNA-seq aligner”. In: *Bioinformatics* 29.1 (2013), pp. 15–21.

[24] Xiaoxi Dong et al. “Reverse enGENEering of Regulatory Networks from Big Data: A Roadmap for Biologists”. In: *Bioinformatics and biology insights* 9 (2015), p. 61.

[25] Franklin H Epstein and David S Latchman. “Transcription-factor mutations and disease”. In: *New England Journal of Medicine* 334.1 (1996), pp. 28–33.

[26] Theresa M Filtz, Walter K Vogel, and Mark Leid. “Regulation of transcription factor activity by interconnected post-translational modifications”. In: *Trends in pharmacological sciences* 35.2 (2014), pp. 76–85.

[27] U Fischer and K Schulze-Osthoff. “Apoptosis-based therapies and drug targets”. In: *Cell Death & Differentiation* 12 (2005), pp. 942–961.

[28] Frank Fontaine, Jeroen Overman, and Mathias FranÃ§ois. “Pharmacological manipulation of transcription factor protein-protein interactions: opportunities and obstacles”. In: *Cell Regeneration* 4.1 (2015), p. 1.

[29] Feng Gao, Barrett C Foat, and Harmen J Bussemaker. “Defining transcriptional networks through integrative modeling of mRNA expression and transcription factor binding data”. In: *BMC bioinformatics* 5.1 (2004), p. 1.

[30] K GÃrtner, H Reznik-SchÃ$\frac{1}{4}$ller, and G Reznik. “The influence of overcrowding on spermatogenesis, size of Leydig-cell nuclei (histometrical investigation), and the adrenal corticosterone contents in mice”. In: *Acta endocrinologica* 74.4 (1973), pp. 783–791.

[31] Geoffrey S Ginsburg and Huntington F Willard. “Genomic and personalized medicine: foundations and applications”. In: *Translational research* 154.6 (2009), pp. 277–287.

[32] Federico M Giorgi, Cristian Del Fabbro, and Francesco Licausi. “Comparative study of RNA-seq-and microarray-derived coexpression networks in Arabidopsis thaliana”. In: *Bioinformatics* 29.6 (2013), pp. 717–724.

[33] Michael V Gormally et al. “Suppression of the FOXM1 transcriptional programme via novel small molecule inhibition”. In: *Nature communications* 5 (2014).

[34] Xiaobo Guo et al. “Inferring nonlinear gene regulatory networks from gene expression data based on distance correlation”. In: *PloS one* 9.2 (2014), e87446.

[35] Douglas Hanahan and Robert A Weinberg. “Hallmarks of cancer: the next generation”. In: *cell* 144.5 (2011), pp. 646–674.

[36] Reina Haque et al. “Impact of breast cancer subtypes and treatment on survival: an analysis spanning two decades”. In: *Cancer Epidemiology Biomarkers & Prevention* 21.10 (2012), pp. 1848–1855.

[37] Tamar Hashimshony et al. “CEL-Seq: single-cell RNA-Seq by multiplexed linear amplification”. In: *Cell reports* 2.3 (2012), pp. 666–673.

[38] Saiful Islam et al. “Characterization of the single-cell transcriptional landscape by highly multiplex RNA-seq”. In: *Genome research* 21.7 (2011), pp. 1160–1167.

[39] Marian Iwamoto et al. “Clinical pharmacology profile of vorinostat, a histone deacetylase inhibitor”. In: *Cancer chemotherapy and pharmacology* 72.3 (2013), pp. 493–508.

[40] Diego Adhemar Jaitin et al. “Massively parallel single-cell RNA-seq for marker-free decomposition of tissues into cell types”. In: *Science* 343.6172 (2014), pp. 776–779.

[41] Y Ji et al. “Inhibition of Autophagy Increases Proliferation Inhibition and Apoptosis Induced by the PI3K/mTOR Inhibitor NVP-BEZ235 in Breast Cancer Cells.” In: *Clinical laboratory* 61.8 (2014), pp. 1043–1051.

[42] Simon J Johnston and Jason S Carroll. “Transcription factors and chromatin proteins as therapeutic targets in cancer”. In: *Biochimica et Biophysica Acta (BBA)-Reviews on Cancer* 1855.2 (2015), pp. 183–192.

[43] Frederick Kinyua Kamanu et al. “Mutations and binding sites of human transcription factors”. In: *Frontiers in genetics* (2012).

[44] Manolis Kellis et al. “Sequencing and comparison of yeast species to identify genes and regulatory elements”. In: *Nature* 423.6937 (2003), pp. 241–254.

[45] Mi-Young Kim et al. “Role of transcription factor modifications in the pathogenesis of insulin resistance”. In: *Experimental diabetes research* 2012 (2011).

[46] Michael S Kinch et al. “An overview of FDA-approved new molecular entities: 1827–2013”. In: *Drug discovery today* 19.8 (2014), pp. 1033–1039.

[47] Brooke LaFlamme. “Molecular barcodes improve RNA-seq”. In: *Nature Genetics* 46.3 (2014), pp. 219–219.

[48] Justin Lamb. “The Connectivity Map: a new tool for biomedical research”. In: *Nature Reviews Cancer* 7.1 (2007), pp. 54–60.

[49] David S Latchman. “Transcription factors: an overview”. In: *The international journal of biochemistry & cell biology* 29.12 (1997), pp. 1305–1312.

[50] Jeffrey T Leek et al. “EDGE: extraction and analysis of differential gene expression”. In: *Bioinformatics* 22.4 (2006), pp. 507–508.

[51] Celine Lefebvre et al. “A human B-cell interactome identifies MYB and FOXM1 as master regulators of proliferation in germinal centers”. In: *Molecular systems biology* 6.1 (2010), p. 377.

[52] Yu Li, Sicong Zhang, and Suyun Huang. “FoxM1: a potential drug target for glioma”. In: *Future Oncology* 8.3 (2012), pp. 223–226.

[53] Yanbin Liu et al. “DCDB: drug combination database”. In: *Bioinformatics* 26.4 (2010), pp. 587–588.

[54] Michael I Love, Wolfgang Huber, and Simon Anders. “Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2”. In: *Genome biology* 15.12 (2014), pp. 1–21.

[55] Ricardo Macarron et al. “Impact of high-throughput screening in biomedical research”. In: *Nature reviews Drug discovery* 10.3 (2011), pp. 188–195.

[56] Adam A Margolin et al. “ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context”. In: *BMC bioinformatics* 7.Suppl 1 (2006), S7.

[57] Eliana Marinari et al. “Live-cell delamination counterbalances epithelial growth to limit tissue overcrowding”. In: *Nature* 484.7395 (2012), pp. 542–545.

[58] Salvatore AE Marras et al. “Real-time measurement of in vitro transcription”. In: *Nucleic acids research* 32.9 (2004), e72–e72.

[59] Bronagh McCabe, Fabio Liberante, and Ken I Mills. “Repurposing medicinal compounds for blood cancer treatment”. In: *Annals of hematology* 94.8 (2015), pp. 1267–1276.

[60] Davis J McCarthy, Yunshun Chen, and Gordon K Smyth. “Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation”. In: *Nucleic acids research* (2012), gks042.

[61] Barbara Melosky et al. “Pointed Progress in Second-Line Advanced Non–Small-Cell Lung Cancer: The Rapidly Evolving Field of Checkpoint Inhibition”. In: *Journal of Clinical Oncology* (2016), JCO638049.

[62] Asher Mullard. “2012 FDA drug approvals”. In: *Nature Reviews Drug Discovery* 12.2 (2013), pp. 87–90.

[63] Chris J Needham et al. “A primer on learning in Bayesian networks for computational biology”. In: *PLoS Comput Biol* 3.8 (2007), e129.

[64] Marjorie F Oleksiak, Gary A Churchill, and Douglas L Crawford. “Variation in gene expression within and among natural populations”. In: *Nature genetics* 32.2 (2002), pp. 261–266.

[65] Antonio Palumbo et al. “Genetic instability in the tumor microenvironment: a new look at an old neighbor”. In: *Molecular cancer* 14.1 (2015), p. 1.

[66] Olivier E Pardo et al. “FGF-2 protects small cell lung cancer cells from apoptosis through a complex involving PKCε, B-Raf and S6K2”. In: *The EMBO journal* 25.13 (2006), pp. 3078–3088.

[67] Staffan Persson et al. “Identification of genes required for cellulose synthesis by regression analysis of public microarray data sets”. In: *Proceedings of the National Academy of Sciences of the United States of America* 102.24 (2005), pp. 8633–8638.

[68] Shahram Jevin Poureetezadi, Eric K Donahue, and Rebecca A Wingert. “A manual small molecule screen approaching high-throughput using zebrafish embryos”. In: *Journal of visualized experiments: JoVE* 93 (2014).

[69] Mary Elizabeth Pownall and Harry V Isaacs. “FGF Ligands”. In: (2010).

[70] Ming Quan et al. “The roles of FOXM1 in pancreatic stem cells and carcinogenesis”. In: *Molecular cancer* 12.1 (2013), p. 1.

[71] VM Richon. “Cancer biology: mechanism of antitumour action of vorinostat (suberoylanilide hydroxamic acid), a novel histone deacetylase inhibitor”. In: *British journal of cancer* 95 (2006), S2–S6.

[72] Nicholas J Roberts et al. “The predictive capacity of personal genome sequencing”. In: *Science translational medicine* 4.133 (2012), 133ra58–133ra58.

[73] Frederick P Roth et al. “Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation”. In: *Nature biotechnology* 16.10 (1998), pp. 939–945.

[74] Roman Rouzier et al. “Breast cancer molecular subtypes respond differently to preoperative chemotherapy”. In: *Clinical Cancer Research* 11.16 (2005), pp. 5678–5685.

[75] Diana Saleiro and Leonidas C Platanias. “Intersection of mTOR and STAT signaling in immunity”. In: *Trends in immunology* 36.1 (2015), pp. 21–29.

[76] Uttara Saran, Michelangelo Foti, and Jean-FranÃ§ois Dufour. “Cellular and molecular effects of the mTOR inhibitor everolimus”. In: *Clinical Science* 129.10 (2015), pp. 895–914.

[77] Alexander A Shishkin et al. “Simultaneous generation of many RNA-seq libraries in a single reaction”. In: *Nature methods* 12.4 (2015), pp. 323–325.

[78] Sergey Shmakov et al. “Discovery and functional characterization of diverse class 2 CRISPR-Cas systems”. In: *Molecular cell* 60.3 (2015), pp. 385–397.

[79] Jose M Silva et al. “Profiling essential genes in human mammary cells by multiplex RNAi screening”. In: *Science* 319.5863 (2008), pp. 617–620.

[80] David Sims et al. “High-throughput RNA interference screening using pooled shRNA libraries and next generation sequencing”. In: *Genome biology* 12.10 (2011), pp. 1–13.

[81] Nikolaos Soulitzis et al. “Expression analysis of peptide growth factors VEGF, FGF2, TGFB1, EGF and IGF1 in prostate cancer and benign prostatic hyperplasia”. In: *International journal of oncology* 29.2 (2006), pp. 305–314.

[82] Patricia M de Souza and Mark A Lindsay. “Apoptosis as a therapeutic target for the treatment of lung disease”. In: *Current opinion in pharmacology* 5.3 (2005), pp. 232–237.

[83] Elmar Stickeler. “Prognostic and predictive markers for treatment decisions in early breast cancer”. In: *Breast Care* 6.3 (2011), pp. 193–198.

[84] Nikolaus J Sucher. “Searching for synergy in silico, in vitro and in vivo”. In: *Synergy* 1.1 (2014), pp. 30–43.

[85] Gergely Szakacs et al. “The role of ABC transporters in drug absorption, distribution, metabolism, excretion and toxicity (ADME–Tox)”. In: *Drug discovery today* 13.9 (2008), pp. 379–393.

[86] Zhiwen Tang et al. “Real-time investigation of nucleic acids phosphorylation process using molecular beacons”. In: *Nucleic acids research* 33.11 (2005), e97–e97.

[87] Sonia Tarazona et al. “Differential expression in RNA-seq: a matter of depth”. In: *Genome research* 21.12 (2011), pp. 2213–2223.

[88] Tina L Tootle and Ilaria Rebay. “Post-translational modifications influence transcription factor activity: A view from the ETS superfamily”. In: *Bioessays* 27.3 (2005), pp. 285–298.

[89] Nicholas Turner and Richard Grose. “Fibroblast growth factor signalling: from development to cancer”. In: *Nature Reviews Cancer* 10.2 (2010), pp. 116–129.

[90] Veerle L Van Marck and Marc E Bracke. “Epithelial-mesenchymal transitions in human cancer”. In: *Rise and Fall of Epithelial Phenotype*. Springer, 2005, pp. 135–159.

[91] Henk MW Verheul. “Mass spectrometry-based proteomics: from cancer biology to protein biomarkers, drug targets, and clinical applications”. In: American Society of Clinical Oncology. 2014.

[92] Nathalie Villa-Vialaneix et al. “The structure of a gene co-expression network reveals biological functions underlying eQTLs”. In: *PloS one* 8.4 (2013), e60045.

[93] Tobias Wagner et al. “Identification of a small-molecule ligand of the epigenetic reader protein Spindlin1 via a versatile screening platform”. In: *Nucleic acids research* (2016), gkw089.

[94] Yi Wang et al. “Function annotation of an SBP-box gene in Arabidopsis based on analysis of co-expression networks and promoters”. In: *International journal of molecular sciences* 10.1 (2009), pp. 116–132.

[95] Thomas Weichhart, Markus HengstschlÃger, and Monika Linke. “Regulation of innate immune cell function by mTOR”. In: *Nature Reviews Immunology* 15.10 (2015), pp. 599–614.

[96] Justin Werfel et al. “How changes in extracellular matrix mechanics and gene expression variability might combine to drive cancer progression”. In: *PloS one* 8.10 (2013), e76122.

[97] Xiaohong R Yang et al. “Differences in risk factors for breast cancer molecular subtypes in a population-based study”. In: *Cancer Epidemiology Biomarkers & Prevention* 16.3 (2007), pp. 439–443.

[98] Jennifer E Yeh, Patricia A Toniolo, and David A Frank. “Targeting transcription factors: promising new strategies for cancer therapy”. In: *Current opinion in oncology* 25.6 (2013), pp. 652–658.

[99] Arabella Young et al. “RNA-seq profiling of a radiation resistant and radiation sensitive prostate cancer cell line highlights opposing regulation of DNA repair and targets for radiosensitization”. In: *BMC cancer* 14.1 (2014), p. 808.

[100] Bai Zhang, Ye Tian, and Zhen Zhang. “Network Biology in Medicine and Beyond”. In: *Circulation: Cardiovascular Genetics* 7.4 (2014), pp. 536–547.

[101] Jianjun Zhang et al. “Intratumor heterogeneity in localized lung adenocarcinomas delineated by multiregion sequencing”. In: *Science* 346.6206 (2014), pp. 256–259.

[102] Xiujun Zhang et al. “Inferring gene regulatory networks from gene expression data by path consistency algorithm based on conditional mutual information”. In: *Bioinformatics* 28.1 (2012), pp. 98–104.

Chapter 6

Appendix

Appendix A Abbreviations Used Throught the Text

ADMET ... Absorption, distribution, metabolism, excretion and toxicity

BN ... Bayesian network

cDNA ... Complementary DNA

DGE ... Differential gene expression

dsDNA ... Double-stranded DNA

GBM ... Glioblastoma

GEP ... Gene expression profile

GRN ... Gene regulatory network

HDAC ... Histone deacetylase

HTS ... High-throughput screening

MoA ... Mechanism of action

MoI ... Multiplicity of infection

mRNA ... messenger RNA

rRNA ... ribosomal RNA

ssDNA ... Single-stranded DNA

Appendix B Drugs Used in Large-Scale Screen and Their Indications

Drug	Indication
Camptothecin	Topoisomerase inhibitor
Doxorubicin	Topoisomerase inhibitor
Clofarabine	Antimetabolite antineoplastic
Topotecan	Topoisomerase inhibitor
Irinotecan	Topoisomerase inhibitor
Epirubicin	Topoisomerase inhibitor
Paclitaxel	Taxane
Adapalene	Retinoid
Cephalomannine	Taxane
Flubendazole	Anthelmintic
Chlorpromazine	Phenothiazine
Daunorubicin	Topoisomerase inhibitor
Dequalinium	Antiseptic
Cetrimonium Bromide	Antiseptic
Oxethazaine	Anaesthetic
Afatinib	EGFR/HER2 inhibitor
9-Aminoacridine	Antiseptic
Teniposide	Topoisomerase inhibitor
Dasatinib	RTK inhibitor
Emetine	Emetic
Digoxigenin	Steroid
Ziprasidone	Antipsychotic
Ouabain	Cardioactive glycoside
BIBR-1048	Thrombin inhibitor
2-Methoxyestradiol	Estradiol metabolite
Etoposide	Topoisomerase inhibitor

Simvastatin	HMG-CoA reductase inhibitor
Gemcitabine	Antineoplastic
Candesartan cilexetil	Angiotensin II receptor antagonist
Sunitinib Malate	RTK inhibitor
Trifluoperazine	Phenothiazine
Bortezomib	Proteasome inhibitor
Albendazole	Anthelmintic
Adefovir	Reverse transcriptase inhibitor
Amlodipine	Ca2+ channel blocker
Estradiol valerate	Contraceptive
Idarubicin	Topoisomerase inhibitor
Fenbendazole	Anthelmintic
Carfilzomib	Proteasome inhibitor
Cabzatinib	VEGFR inhibitor
Nilvadipine	Ca2+ channel blocker
Temsirolimus	mTOR inhibitor
Triflupromazine	Antipsychotic
Amlodipine	Ca2+ channel blocker
Fosaprepitant	Antiemetic
Cetylpyridinium	Antimicrobial
AZD7762	Chk2 inhibitor
LDK378	ALK inhibitor
AZD3463	ALK inhibitor
BKM120	PI3K inhibitor
Nilotinib	RTK inhibitor
BGJ398	FGFR kinase inhibitor
Ponatinib	RTK inhibitor
Ivacaftor	CFTR potentiator
AMG-073	Calcimimetic
Disulfiram	ADH inhibitor

Crizotinib	RTK inhibitor
Aprepitant	Antiemetic
Thioguanine	Antineoplastic
Sorafenib	RTK inhibitor
Ivermectin	Anthelmintic
Nebivolol	Antihypertensive
Amorolfine	Antifungal
Chloroxine	Antibacterial
Dyclonine	Anaesthetic
Crystal Violet	Dye
Mitoxantrone	Topoisomerase inhibitor

Appendix C Examples of Code Used in Analyses

C.1 Differential Gene Expression Analysis

```
1
2 library(DESeq2)
3
4 load("smallScreen_truseq_deseqObj.rda")
5 load("smallScreen_plateseq_deseqObj.rda")
6
7 # Get DGE results & place into an object
8 # PLATE-Seq
9 res <- results(dds)
10 table(res$padj<0.05)
11 ## Order by adjusted p-value
12 res <- res[order(res$padj), ]
13 ## Merge with normalized count data
14 resdata <- merge(as.data.frame(res), as.data.frame(counts(dds, normalized=TRUE
      )), by="row.names", sort=FALSE)
15 names(resdata)[1] <- "Gene"
16
17 # TruSeq
18 res2 <- results(dds2)
19 table(res2$padj<0.05)
```

```
## Order by adjusted p-value
res2 <- res2[order(res2$padj), ]
## Merge with normalized count data
resdata2 <- merge(as.data.frame(res2), as.data.frame(counts(dds2, normalized=
    TRUE)), by="row.names", sort=FALSE)
names(resdata2)[1] <- "Gene"

# MA plot, PLATE-Seq
maplot <- function (res, thresh=0.05, labelsig=TRUE, textcx=1, ...) {
  with(res, plot(baseMean, log2FoldChange, pch=20, cex=.5, log="x"), ...)
  with(subset(res, padj<thresh), points(baseMean, log2FoldChange, col="red",
    pch=20, cex=1.5))
  if (labelsig) {
    require(calibrate)
    with(subset(res, padj<thresh), textxy(baseMean, log2FoldChange, labs=Gene,
     cex=textcx, col=2))
  }
}

# MA plot, TruSeq
maplot2 <- function (res2, thresh=0.05, labelsig=TRUE, textcx=1, ...) {
  with(res2, plot(baseMean, log2FoldChange, pch=20, cex=.5, log="x", ...))
  with(subset(res2, padj<thresh), points(baseMean, log2FoldChange, col="red",
    pch=20, cex=1.5))
  if (labelsig) {
    require(calibrate)
    with(subset(res2, padj<thresh), textxy(baseMean, log2FoldChange, labs=Gene
    , cex=textcx, col=2))
  }
}

# Volcano plot, PLATE-Seq
volcanoplot <- function (res, lfcthresh=2, sigthresh=0.05, main="Volcano Plot"
    , legendpos="bottomright", labelsig=TRUE, textcx=1, ...) {
  with(res, plot(log2FoldChange, -log10(pvalue), pch=20, main=main, ...))
  with(subset(res, padj<sigthresh ), points(log2FoldChange, -log10(pvalue),
    pch=20, col="red", ...))
  with(subset(res, abs(log2FoldChange)>lfcthresh), points(log2FoldChange, -
    log10(pvalue), pch=20, col="orange", ...))
  with(subset(res, padj<sigthresh & abs(log2FoldChange)>lfcthresh), points(
    log2FoldChange, -log10(pvalue), pch=20, col="green", ...))
  if (labelsig) {
    require(calibrate)
    with(subset(res, padj<sigthresh & abs(log2FoldChange)>lfcthresh), textxy(
    log2FoldChange, -log10(pvalue), labs=Gene, cex=textcx, ...))
  }
  legend(legendpos, xjust=1, yjust=1, legend=c(paste("FDR<",sigthresh,sep=""),
     paste("|LogFC|>",lfcthresh,sep=""), "both"), pch=20, col=c("red","orange"
```

```
      ,"green"))
}

# Volcano plot, TruSeq
volcanoplot2 <- function (res2, lfcthresh=2, sigthresh=0.05, main="Volcano
    Plot", legendpos="bottomright", labelsig=TRUE, textcx=1, ...) {
  with(res2, plot(log2FoldChange, -log10(pvalue), pch=20, main=main, ...))
  with(subset(res2, padj<sigthresh ), points(log2FoldChange, -log10(pvalue),
    pch=20, col="red", ...))
  with(subset(res2, abs(log2FoldChange)>lfcthresh), points(log2FoldChange, -
    log10(pvalue), pch=20, col="orange", ...))
  with(subset(res2, padj<sigthresh & abs(log2FoldChange)>lfcthresh), points(
    log2FoldChange, -log10(pvalue), pch=20, col="green", ...))
  if (labelsig) {
    require(calibrate)
    with(subset(res2, padj<sigthresh & abs(log2FoldChange)>lfcthresh), textxy(
    log2FoldChange, -log10(pvalue), labs=Gene, cex=textcx, ...))
  }
  legend(legendpos, xjust=1, yjust=1, legend=c(paste("FDR<",sigthresh,sep=""),
     paste("|LogFC|>",lfcthresh,sep=""), "both"), pch=20, col=c("red","orange"
    ,"green"))
}

# Plot QC measures
pdf("DESeq2_QC_Plots.pdf", w=12, h=18, pointsize = 18)
par(mfrow=c(3,2))
plotDispEsts(dds, main="PLATE_Seq Dispersions, Undetected Genes Removed")
plotDispEsts(dds2, main="TruSeq Dispersions, Undetected Genes Removed")

maplot(resdata, main="PLATE-Seq MA")
maplot(resdata2, main="TruSeq MA")

volcanoplot(resdata, lfcthresh=1, sigthresh=0.05, textcx=.8, xlim=c(-2.3, 2),
    main="DGE, PLATE-Seq")
volcanoplot(resdata2, lfcthresh=1, sigthresh=0.05, textcx=.8, xlim=c(-2.3, 2),
     main="DGE, TruSeq")
dev.off()
```

Algorithm 6.1: Estimate differential gene expression using DESeq2

C.2 Computing and Plotting t-SNE

```

# Multidimensional scaling of the N-of-1 data

```

```
# load gene expression signatures
load("~/Projects/PLATE-Seq/N1/BreastCancerScreen/nof1_bctx_normalizedCts.rda")
# load viper object
load("~/Projects/PLATE-Seq/N1/BreastCancerScreen/nof1_vipermat.rda")

# TSNE
library(tsne)
# separate signature objects by time point
gep6 <- s1[,c(1:92,185:276,369:460)]
gep24 <- s1[,c(93:184,277:368,461:552)]

vp6 <- vp1[,c(1:92,185:276,369:460)]
vp24 <- vp1[,c(93:184,277:368,461:552)]

tgep6 <- tsne(t(gep6))
tgep24 <- tsne(t(gep24))
tvp6 <- tsne(t(vp6))
tvp24 <- tsne(t(vp24))

pdf("bt20_tsne_allSamples_withLabels.pdf", h=20, w=20, pointsize=18)
par(mfrow=c(2,2))
plot(tgep6, pch=20, col=rgb(0,0,0,0.5), xlab="Projection 1", ylab="Projection
    2", main="TSNE of Gene Expression Profiles, 6hr")
text(tgep6[,1], tgep6[,2], labels=colnames(gep6), cex=0.4)

plot(tgep24, pch=20, col=rgb(0,0,0,0.5), xlab="Projection 1", ylab="Projection
     2", main="TSNE of Gene Expression Profiles, 24hr")
text(thitgep24ng[,1], tgep24[,2], labels=colnames(gep24), cex=0.4)

plot(tvp6, pch=20, col=rgb(0,0,0,0.5), xlab="Projection 1", ylab="Projection 2
    ", main="TSNE of VIPER Profiles, 6hr")
text(tvp6[,1], tvp6[,2], labels=colnames(vp6), cex=0.4)

plot(tvp24, pch=20, col=rgb(0,0,0,0.5), xlab="Projection 1", ylab="Projection
    2", main="TSNE of VIPER Profiles, 24hr")
text(tvp24[,1], tvp24[,2], labels=colnames(vp24), cex=0.4)

dev.off()

##########################
# Single plate analysis
gPlate1_6 <- s1[,c(1:92)]
gPlate1_6 <- gPlate1_6[,order(colnames(gPlate1_6))]
gPlate1_24 <- s1[,c(93:184)]
gPlate1_24 <- gPlate1_24[,order(colnames(gPlate1_24))]

vPlate1_6 <- vp1[,c(1:92)]
vPlate1_6 <- vPlate1_6[,order(colnames(vPlate1_6))]
```

```
vPlate1_24 <- vp1[,c(93:184)]
vPlate1_24 <- vPlate1_24[,order(colnames(vPlate1_24))]

tplate1_6 <- tsne(t(gPlate1_6))
tplate1_24 <- tsne(t(gPlate1_24))
tviper1_6 <- tsne(t(vPlate1_6))
tviper1_24 <- tsne(t(vPlate1_24))

pdf("bt20_tsne_singlePlate_allSamples_withLabels.pdf", h=20, w=20, pointsize
    =18)
par(mfrow=c(2,2))
plot(tplate1_6, pch = seq(1,23, by=2), col=rgb(0,0,0,0.5), xlab="Projection 1"
    , ylab="Projection 2", main="TSNE of Gene Expression Profiles, 6hr")
# text(tplate1_6[,1], tplate1_6[,2], labels=colnames(gPlate1_6), cex=0.4)

plot(tplate1_24, pch = seq(1,23, by=2), col=rgb(0,0,0,0.5), xlab="Projection 1
    ", ylab="Projection 2", main="TSNE of Gene Expression Profiles, 24hr")
# text(tplate1_24[,1], tplate1_24[,2], labels=colnames(gPlate1_24), cex=0.4)

plot(tviper1_6, pch = seq(1,23, by=2), col=rgb(0,0,0,0.5), xlab="Projection 1"
    , ylab="Projection 2", main="TSNE of VIPER Profiles, 6hr")
# text(tviper1_6[,1], tviper1_6[,2], labels=colnames(vPlate1_6), cex=0.4)

plot(tviper1_24, pch = seq(1,23, by=2), col=rgb(0,0,0,0.5), xlab="Projection 1
    ", ylab="Projection 2", main="TSNE of VIPER Profiles, 24hr")
# text(tviper1_24[,1], tviper1_24[,2], labels=colnames(vPlate1_24), cex=0.4)

dev.off()
```

Algorithm 6.2: Computing t-SNE and plotting the results.

C.3 Calculating Gene Expression and VIPER Activity Signatures

```

setwd("~/Projects/PLATE-Seq/exo/Expt2_exoPlusOnly/norm2All/")
packages <- c("viper", "atools", "aanot", "DESeq", "limma", "dplyr", "DeMAND",
    "gplots")
sapply(packages, require, character.only = TRUE, quietly = TRUE)
rm(packages)
source("~/Projects/PLATE-Seq/exo/Expt2_exoPlusOnly/normToDMSO/viperSimilarity2
    .R")

load("../TruSeq/PeterCounts/truseq_rawCounts_Peter.rda")     # truseq RAW
    counts
load("../deExoRepeat.RData")                                 # plateseq DE-
    stabilized counts
```

```
10
11 deTS <- DEtransform(truseqRaw)
12 rm(truseqRaw)
13
14 # Check the correlations between treatments
15 tsne(deTS[1:1000,])
16
17 # dePlus <- dePlus[, -grep("_1\$", colnames(dePlus))]
18 dePlus <- dePlus[,-which(colnames(dePlus)=="DMSO_5")]
19 pcplot(dePlus)
20 ### Computing signatures relative to DMSO for each sample #######
21 #assign columns labeled "DMSO" as ctrl
22 tx <- t(sapply(strsplit(colnames(dePlus), "_"), function(x) x))
23 ctrl <- dePlus[, tx[,1]=="DMSO"]
24
25 # Create a matrix holding only treatment samples
26 treat <- dePlus[, tx[,1]!="DMSO"]
27
28 # Repeat above steps for TruSeq library
29 tx2 <- t(sapply(strsplit(colnames(deTS), "_"), function(x) x))
30 ctrl2 <- deTS[, tx2[,1]=="DMSO"]
31
32 # Create a matrix holding only treatment samples
33 treat30 <- deTS[, tx2[,1]!="DMSO"]
34
35 # Normalize GES for each replicate against all others
36 tsCmpAll <- t(scale(t(treat30))) # subtracts the mean of each row and divides
       by the standard deviation of each row
37 colnames(tsCmpAll) <- paste("TruSeq", colnames(tsCmpAll), sep = "_")
38 save(tsCmpAll, file="GES_norm2All_TruSeq.rda")
39 psCmpAll <- t(scale(t(treat)))
40 colnames(psCmpAll) <- paste("PLATESeq", colnames(psCmpAll), sep = "_")
41 save(psCmpAll, file="GES_norm2All_PLATESEQ.rda")
42
43 # Sort expression matrices by treatment name & combine matrices
44 psCmpAll <- psCmpAll[,sort(colnames(psCmpAll))]
45 tsCmpAll <- tsCmpAll[,sort(colnames(tsCmpAll))]
46 genes <- intersect(rownames(psCmpAll), rownames(tsCmpAll))
47 gepcmp <- cbind(psCmpAll[match(genes, rownames(psCmpAll)), ], tsCmpAll[match(
       genes, rownames(tsCmpAll)), ])
48
49 # Test treatment replicate similarities
50 dd1 <- viperSimilarity(gepcmp, nn=50, method="two.sided")
51 par(mar=c(1, 10, 12, 1) + 0.1)
52 diag(dd1) <- NA
53 plothm(dd1, grid=F)
54 axis(3, 1:ncol(gepcmp), colnames(gepcmp), tick=FALSE, line=-.5, las=2, cex.
       axis=0.7)
```

```
axis(2, ncol(gepcmp):1, colnames(gepcmp), tick=FALSE, line=-.5, las=2, cex.axis=0.7)
title(main="A. GEP Similarity Between Treatment Replicates", line=10, cex.main=2)

# Plot PLATE-Seq samples against TruSeq samples by gene expression (84x14)
# vsm <- dd1[1:77,][,-c(1:77)]
vsm <- dd1[1:84,][,-c(1:84)]
par(mar=c(1, 10, 12, 1) + 0.1)
plothm(vsm, grid=F)
# rect(c(1,3,5,7,9,11,13)-0.5, c(67,56,45,34,23,12,1)-0.5, c(2,4,6,8,10,12,14)+0.5, c(77,66,55,44,33,22,11)+0.5, lwd=5)
rect(c(1,3,5,7,9,11,13)-0.5, c(73,61,49,37,25,13,1)-0.5, c(2,4,6,8,10,12,14)+0.5, c(84,72,60,48,36,24,12)+0.5, lwd=5)
axis(3, 1:ncol(vsm), colnames(vsm), tick=FALSE, line=-.5, las=2, cex.axis=0.7)
axis(2, nrow(vsm):1, rownames(vsm), tick=FALSE, line=-.5, las=2, cex.axis=0.7)
title(main="B. GEP Similarity Between Library Preparation Methods", line=10, cex.main=2)

# VIPER
# BRCA TCGA regulon
load("~/Projects/PLATE-Seq/Interactomes/brca-tcga-tf-regulon.rda")
reg <- regul
rm(reg)

vps <- viper(psCmpAll, regul, method="none")
save(vps, file="viperRes_norm2All_PLATESEQ.rda")

vts <- viper(tsCmpAll, regul, method="none")
save(vts, file="viperRes_norm2All_TruSeq.rda")

tfs <- intersect(rownames(vps), rownames(vts))
tfcmp <- cbind(vps[match(tfs, rownames(vps)), ], vts[match(tfs, rownames(vts)), ])

# Plot PLATE-Seq samples against TruSeq samples by VIPER activity (84x14)
dd2 <- scale(viperSimilarity2(tfcmp, nn=0.01, method="two.sided"))  # values in this plot have to be scaled because of the large difference in NES between TruSeq & PLATE-Seq, stemming from the different gene set sizes
par(mar=c(1, 10, 12, 1) + 0.1)
diag(dd2) <- NA
plothm(dd2, grid=F)
axis(3, 1:ncol(tfcmp), colnames(tfcmp), tick=FALSE, line=-.5, las=2, cex.axis=0.7)
axis(2, ncol(tfcmp):1, colnames(tfcmp), tick=FALSE, line=-.5, las=2, cex.axis=0.7)
title(main="C. VIPER Similarity Between Treatment Replicates", line=10, cex.main=2)
```

```
92
93 # Plot PLATE-Seq samples against TruSeq samples on a tx-tx- basis (12x2)
94 vsm2 <- dd2[1:84,][,-c(1:84)]
95 par(mar=c(1, 10, 12, 1) + 0.1)
96 plothm(vsm2, grid=F)
97 rect(c(1,3,5,7,9,11,13)-0.5, c(73,61,49,37,25,13,1)-0.5, c(2,4,6,8,10,12,14)
       +0.5, c(84,72,60,48,36,24,12)+0.5, lwd=5)
98 # rect(c(1,3,5,7,9,11,13)-0.5, c(67,56,45,34,23,12,1)-0.5, c(2,4,6,8,10,12,14)
       +0.5, c(77,66,55,44,33,22,11)+0.5, lwd=5)
99 axis(3, 1:ncol(vsm2), colnames(vsm2), tick=FALSE, line=-.5, las=2, cex.axis
       =0.7)
100 axis(2, nrow(vsm2):1, rownames(vsm2), tick=FALSE, line=-.5, las=2, cex.axis
       =0.7)
101 title(main="D. VIPER Similarity Between Library Preparation Methods", line=10,
        cex.main=2)
102
103
104 # Plot bloody everything together
105 pdf("Comparatron_outliersRemoved_norm2All_BRCA_TFonly.pdf", w=32, h=32,
       pointsize=20)
106 par(mfrow=c(2,2))
107 par(mar=c(1, 10, 12, 1) + 0.1)
108 plothm(dd1, grid=F)
109 axis(3, 1:ncol(gepcmp), colnames(gepcmp), tick=FALSE, line=-.5, las=2, cex.
       axis=0.7)
110 axis(2, ncol(gepcmp):1, colnames(gepcmp), tick=FALSE, line=-.5, las=2, cex.
       axis=0.7)
111 title(main="A. GEP Similarity Between Treatment Replicates", line=10, cex.main
       =2)
112
113 plothm(vsm, grid=F)
114 rect(c(1,3,5,7,9,11,13)-0.5, c(73,61,49,37,25,13,1)-0.5, c(2,4,6,8,10,12,14)
       +0.5, c(84,72,60,48,36,24,12)+0.5, lwd=5)
115 axis(3, 1:ncol(vsm), colnames(vsm), tick=FALSE, line=-.5, las=2, cex.axis=0.7)
116 axis(2, nrow(vsm):1, rownames(vsm), tick=FALSE, line=-.5, las=2, cex.axis=0.7)
117 title(main="B. GEP Similarity Between Library Preparation Methods", line=10,
       cex.main=2)
118
119 plothm(dd2, grid=F)
120 axis(3, 1:ncol(tfcmp), colnames(tfcmp), tick=FALSE, line=-.5, las=2, cex.axis
       =0.7)
121 axis(2, ncol(tfcmp):1, colnames(tfcmp), tick=FALSE, line=-.5, las=2, cex.axis
       =0.7)
122 title(main="C. VIPER Similarity Between Treatment Replicates", line=10, cex.
       main=2)
123
124 plothm(vsm2, grid=F)
```

```
rect(c(1,3,5,7,9,11,13)-0.5, c(73,61,49,37,25,13,1)-0.5, c(2,4,6,8,10,12,14)
    +0.5, c(84,72,60,48,36,24,12)+0.5, lwd=5)
axis(3, 1:ncol(vsm2), colnames(vsm2), tick=FALSE, line=-.5, las=2, cex.axis
    =0.7)
axis(2, nrow(vsm2):1, rownames(vsm2), tick=FALSE, line=-.5, las=2, cex.axis
    =0.7)
title(main="D. VIPER Similarity Between Library Preparation Methods", line=10,
     cex.main=2)
dev.off()

# Plot the two 12x84 graphs together
pdf("PSvsTS_norm2All_BRCA_TFonly.pdf", w=32, h=16, pointsize=20)
par(mfrow=c(1,2))
par(mar=c(1, 10, 12, 1) + 0.1)
plothm(vsm, grid=F)
rect(c(1,3,5,7,9,11,13)-0.5, c(73,61,49,37,25,13,1)-0.5, c(2,4,6,8,10,12,14)
    +0.5, c(84,72,60,48,36,24,12)+0.5, lwd=5)
axis(3, 1:ncol(vsm), colnames(vsm), tick=FALSE, line=-.5, las=2, cex.axis=0.7)
axis(2, nrow(vsm):1, rownames(vsm), tick=FALSE, line=-.5, las=2, cex.axis=0.7)
title(main="A. GEP Similarity Between Library Preparation Methods", line=10,
    cex.main=1)

plothm(vsm2, grid=F)
rect(c(1,3,5,7,9,11,13)-0.5, c(73,61,49,37,25,13,1)-0.5, c(2,4,6,8,10,12,14)
    +0.5, c(84,72,60,48,36,24,12)+0.5, lwd=5)
axis(3, 1:ncol(vsm2), colnames(vsm2), tick=FALSE, line=-.5, las=2, cex.axis
    =0.7)
axis(2, nrow(vsm2):1, rownames(vsm2), tick=FALSE, line=-.5, las=2, cex.axis
    =0.7)
title(main="B. VIPER Similarity Between Library Preparation Methods", line=10,
     cex.main=1)
dev.off()
```

Algorithm 6.3: Calculating Gene Expression and VIPER Activity Signatures.

C.4 DESeq Normalization and Clustering by MDS

```
# DGE by DESeq2 for TruSeq counts2
setwd("~/Projects/PLATE-Seq/exo/Analysis/")
library(DESeq2)

# Step 1: Differential Gene Expression using DESeq2
# load("~/Projects/PLATE-Seq/exo/Expt2_exoPlusOnly/TruSeq/Petercounts2/truseq_
    rawcounts2_Peter.rda")
counts2 <- read.table("~/Projects/PLATE-Seq/exo/Expt2_exoPlusOnly/TruSeq/
    Petercounts/FR_truseq.cts.txt", sep="\t", header=T)
```

```
rownames(counts2) <- counts2[,1]
counts2 <- counts2[,-1]

colnames(counts2) <- c("Mitoxantrone_1", "Mitoxantrone_2", "Crizotinib_1", "Crizotinib_2",
                       "Albendazole_1", "Albendazole_2", "Idarubicin_1", "Idarubicin_2",
                       "Aprepitant_1", "Aprepitant_2", "Gemcitabine_1", "Gemcitabine_2",
                       "Bortezomib_1", "Bortezomib_2", "DMSO_1", "DMSO_2")

# order columns by tx name
counts2 <- counts2[,order(colnames(counts2))]

# reorder columns to group by treatment & control
counts2 <- counts2[,c(1:8,11:16,9,10)]

# Convert to matrix
counts2 <- as.matrix(counts2)
counts2[1:2,]

# remove rows containing only zeros
counts2 <- counts2[!!rowSums(abs(counts2)),]
save(counts2, file="truseq_rawCounts_sorted.rda")

tx <- t(sapply(strsplit(colnames(counts2), "_"), function(x) x))
dmso <- counts2[, tx[,1]=="DMSO"]
alben <- counts2[, tx[,1]=="Albendazole"]
alben <- cbind(dmso, alben)
(condition1 <- factor(c(rep("DMSO", 2), rep("Albendazole", 2))))

aprep <- counts2[, tx[,1]=="Aprepitant"]
aprep <- cbind(dmso, aprep)
(condition2 <- factor(c(rep("DMSO", 2), rep("Aprepitant", 2))))

bort <- counts2[, tx[,1]=="Bortezomib"]
bort <- cbind(dmso, bort)
(condition3 <- factor(c(rep("DMSO", 2), rep("Bortezomib", 2))))

criz <- counts2[, tx[,1]=="Crizotinib"]
criz <- cbind(dmso, criz)
(condition4 <- factor(c(rep("DMSO", 2), rep("Crizotinib", 2))))

gem <- counts2[, tx[,1]=="Gemcitabine"]
gem <- cbind(dmso, gem)
(condition5 <- factor(c(rep("DMSO", 2), rep("Gemcitabine", 2))))

ida <- counts2[, tx[,1]=="Idarubicin"]
```

```
ida <- cbind(dmso, ida)
(condition6 <- factor(c(rep("DMSO", 2), rep("Idarubicin", 2))))

mito <- counts2[, tx[,1]=="Mitoxantrone"]
mito <- cbind(dmso, mito)
(condition7 <- factor(c(rep("DMSO", 2), rep("Mitoxantrone", 2))))

# Analysis with DESeq2 ------------------------------------------------------

# Create a coldata frame and instantiate the DESeqDataSet. See ?
    DESeqDataSetFromMatrix
(coldata1 <- data.frame(row.names=colnames(alben), condition1))
(coldata2 <- data.frame(row.names=colnames(aprep), condition2))
(coldata3 <- data.frame(row.names=colnames(bort), condition3))
(coldata4 <- data.frame(row.names=colnames(criz), condition4))
(coldata5 <- data.frame(row.names=colnames(gem), condition5))
(coldata6 <- data.frame(row.names=colnames(ida), condition6))
(coldata7 <- data.frame(row.names=colnames(mito), condition7))

dds2_1 <- DESeqDataSetFromMatrix(countData=alben, colData=coldata1, design=~
    condition1)
dds2_2 <- DESeqDataSetFromMatrix(countData=aprep, colData=coldata2, design=~
    condition2)
dds2_3 <- DESeqDataSetFromMatrix(countData=bort, colData=coldata3, design=~
    condition3)
dds2_4 <- DESeqDataSetFromMatrix(countData=criz, colData=coldata4, design=~
    condition4)
dds2_5 <- DESeqDataSetFromMatrix(countData=gem, colData=coldata5, design=~
    condition5)
dds2_6 <- DESeqDataSetFromMatrix(countData=ida, colData=coldata6, design=~
    condition6)
dds2_7 <- DESeqDataSetFromMatrix(countData=mito, colData=coldata7, design=~
    condition7)
# dds2


# Run DESeq
dds2_1 <- DESeq(dds2_1)
dds2_2 <- DESeq(dds2_2)
dds2_3 <- DESeq(dds2_3)
dds2_4 <- DESeq(dds2_4)
dds2_5 <- DESeq(dds2_5)
dds2_6 <- DESeq(dds2_6)
dds2_7 <- DESeq(dds2_7)

mclisty <- list(dds2_1, dds2_2, dds2_3, dds2_4, dds2_5, dds2_6, dds2_7)

resList <- list()
```

```
for(i in 1:7){
  res1 <- results(mclisty[[i]])
  table(res1$padj<0.05)
  res1 <- res1[order(res1$padj), ]
  tmpdat <- merge(as.data.frame(res1), as.data.frame(counts(mclisty[[i]],
    normalized=TRUE)), by="row.names", sort=FALSE)
  names(tmpdat)[1] <- "Gene"
  tmp <- tmpdat[,10:11]
  rownames(tmp) <- tmpdat[,1]
  resList[[length(resList)+1]] <- tmp
}

for(i in 1:7){
  resList[[i]] <- resList[[i]][rownames(resList[[1]]), ] # orders list
    elements by gene ordering of the first table
}

diffGenes <- do.call(cbind, resList)
save(diffGenes, file="DEG_TruSeq.rda")

######################################################################
# Repeat that whole goddam thing for PLATE-Seq
counts <- read.table("PF003_R2.cts.txt")
rownames(counts) <- counts[,1]
counts <- counts[,-1]

# Apply treatment names to columns
tmp <- readLines("exoSamples.txt")  #samples are listed column-wise
colnames(counts) <- tmp
colnames(counts) <- paste0(colnames(counts), "_", rep(1:12, each=8))

# Remove DMSO_5
counts <- counts[,-40]

# Remove ERCC rows
counts <- counts[-grep("ERCC-", rownames(counts)),]

# order columns by tx name
counts <- counts[,order(colnames(counts))]
counts <- counts[,c(1:48,60:95,49:59)]

# Convert to matrix
counts <- as.matrix(counts)
head(counts)
save(counts, file="plateseqRawCounts.rda")

tx <- t(sapply(strsplit(colnames(counts), "_"), function(x) x))
dmso <- counts[, tx[,1]=="DMSO"]
```

```
alben <- counts[, tx[,1]=="Albendazole"]
alben <- cbind(dmso, alben)
(condition1 <- factor(c(rep("DMSO", 11), rep("Albendazole", 12))))

aprep <- counts[, tx[,1]=="Aprepitant"]
aprep <- cbind(dmso, aprep)
(condition2 <- factor(c(rep("DMSO", 11), rep("Aprepitant", 12))))

bort <- counts[, tx[,1]=="Bortezomib"]
bort <- cbind(dmso, bort)
(condition3 <- factor(c(rep("DMSO", 11), rep("Bortezomib", 12))))

criz <- counts[, tx[,1]=="Crizotinib"]
criz <- cbind(dmso, criz)
(condition4 <- factor(c(rep("DMSO", 11), rep("Crizotinib", 12))))

gem <- counts[, tx[,1]=="Gemcitabine"]
gem <- cbind(dmso, gem)
(condition5 <- factor(c(rep("DMSO", 11), rep("Gemcitabine", 12))))

ida <- counts[, tx[,1]=="Idarubicin"]
ida <- cbind(dmso, ida)
(condition6 <- factor(c(rep("DMSO", 11), rep("Idarubicin", 12))))

mito <- counts[, tx[,1]=="Mitoxantrone"]
mito <- cbind(dmso, mito)
(condition7 <- factor(c(rep("DMSO", 11), rep("Mitoxantrone", 12))))

# Analysis with DESeq2 ----------------------------------------------------
# Create a coldata frame and instantiate the DESeqDataSet. See ?
    DESeqDataSetFromMatrix
(coldata1 <- data.frame(row.names=colnames(alben), condition1))
(coldata2 <- data.frame(row.names=colnames(aprep), condition2))
(coldata3 <- data.frame(row.names=colnames(bort), condition3))
(coldata4 <- data.frame(row.names=colnames(criz), condition4))
(coldata5 <- data.frame(row.names=colnames(gem), condition5))
(coldata6 <- data.frame(row.names=colnames(ida), condition6))
(coldata7 <- data.frame(row.names=colnames(mito), condition7))

dds_1 <- DESeqDataSetFromMatrix(countData=alben , colData=coldata1 , design=~
    condition1)
dds_2 <- DESeqDataSetFromMatrix(countData=aprep , colData=coldata2 , design=~
    condition2)
dds_3 <- DESeqDataSetFromMatrix(countData=bort , colData=coldata3 , design=~
    condition3)
dds_4 <- DESeqDataSetFromMatrix(countData=criz , colData=coldata4 , design=~
    condition4)
```

```
dds_5 <- DESeqDataSetFromMatrix(countData=gem, colData=coldata5, design=~
    condition5)
dds_6 <- DESeqDataSetFromMatrix(countData=ida, colData=coldata6, design=~
    condition6)
dds_7 <- DESeqDataSetFromMatrix(countData=mito, colData=coldata7, design=~
    condition7)

# Run DESeq
dds_1 <- DESeq(dds_1)
dds_2 <- DESeq(dds_2)
dds_3 <- DESeq(dds_3)
dds_4 <- DESeq(dds_4)
dds_5 <- DESeq(dds_5)
dds_6 <- DESeq(dds_6)
dds_7 <- DESeq(dds_7)
# The problem now is that there are a lot of NAs in the resulting DESeq
    oblects
mclisty <- list(dds_1, dds_2, dds_3, dds_4, dds_5, dds_6, dds_7)

# Check differential expression results (sanity check)
tmp <- results(dds_2)
table(tmp$padj<0.05)
## Order by adjusted p-value
tmp <- tmp[order(tmp$padj), ]
## Merge with normalized count data
tmpdata <- merge(as.data.frame(tmp), as.data.frame(counts(dds_2, normalized=
    TRUE)), by="row.names", sort=FALSE)
names(tmpdata)[1] <- "Gene"
head(tmpdata)

resList <- list()
for(i in 1:7){
  res1 <- results(mclisty[[i]])
  table(res1$padj<0.05)
  res1 <- res1[order(res1$padj), ]
  tmpdat <- merge(as.data.frame(res1), as.data.frame(counts(mclisty[[i]],
    normalized=TRUE)), by="row.names", sort=FALSE)
  names(tmpdat)[1] <- "Gene"
  tmp <- tmpdat[,19:30]
  rownames(tmp) <- tmpdat[,1]
  resList[[length(resList)+1]] <- tmp
}

for(i in 1:7){
  resList[[i]] <- resList[[i]][rownames(resList[[1]]), ] # orders list
    elements by gene ordering of the first table
}

```

```
diffGenes_ps <- do.call(cbind, resList)
save(diffGenes_ps, file="DEG_plateseq.rda")

# Alternate method: assign DEG by log2 fold change > 2
lfcList <- list()
for(i in 1:7){
  res1 <- results(mclisty[[i]])
  table(abs(res1$log2FoldChange) > 2)
  res1 <- res1[order(res1$padj), ]
  tmpdat <- merge(as.data.frame(res1), as.data.frame(counts(mclisty[[i]],
    normalized=TRUE)), by="row.names", sort=FALSE)
  names(tmpdat)[1] <- "Gene"
  tmp <- tmpdat[,19:30]
  rownames(tmp) <- tmpdat[,1]
  lfcList[[length(lfcList)+1]] <- tmp
}

for(i in 1:7){
  lfcList[[i]] <- lfcList[[i]][rownames(lfcList[[1]]), ] # orders list
    elements by gene ordering of the first table
}

diffGenByLFC_ps <- do.call(cbind, lfcList)
save(diffGenByLFC_ps, file="DEG_byLFC_plateseq.rda")

######################################################################
# Compute Euclidean distances between columns
# PLATE-Seq
sampleDists <- as.matrix(dist(t(diffGenes_ps)))

# TruSeq
sampleDists2 <- as.matrix(dist(t(diffGenes)))

# Plot MDS
library(RColorBrewer)
library(limma)
# Assign condition
(condition <- factor(c(rep("Albendazole", 12), rep("Aprepitant", 12), rep("
    Bortezomib", 12),
                       rep("Crizotinib", 12), rep("Gemcitabine", 12), rep("
    Idarubicin", 12),
                       rep("Mitoxantrone", 12))))

(condition2 <- factor(c(rep("Albendazole", 2), rep("Aprepitant", 2), rep("
    Bortezomib", 2),
                        rep("Crizotinib", 2), rep("Gemcitabine", 2), rep("
    Idarubicin", 2),
                        rep("Mitoxantrone", 2))))
```

```
264
265 (mycols <- brewer.pal(7, "Dark2")[1:length(unique(condition))])
266 (mycols2 <- brewer.pal(7, "Dark2")[1:length(unique(condition2))])
267
268 p1<-plotMDS(as.matrix(sampleDists), top=500, pch=20, col=mycols[condition])
269 p2<-plotMDS(as.matrix(sampleDists2), top=500, pch=0, col=mycols2[condition2])
270
271
272 pdf("mds_plateseqVStruseq.pdf", h=6, w=5, pointsize=12)
273 plot(p2$x,p2$y,pch=0,col=mycols2[condition2], main="MDS of Differential Gene
      Expression")
274 points(p1$x,p1$y,pch=20,col=mycols[condition])
275 dev.off()
276
277 ###############################################################
278 # VIPER analysis of DESeq-analyzed DEGs
279 setwd("~/Projects/PLATE-Seq/exo/Analysis/")
280 library(viper)
281 library(aanot)
282 data(desc)
283
284 load("~/Projects/PLATE-Seq/Interactomes/brca-tcga-tf-regulon.rda")
285 reg <- regul
286 rm(reg)
287
288 # NB: the diffGenes objects were poorly named - they are NORMALIZED, but not
      necessarily differentially expressed. FML.
289 # PLATE-Seq
290 tmp <- gene2entrez(rownames(diffGenes_ps))
291
292 unique_indices <- which(!duplicated(tmp) & !is.na(tmp))
293
294 tmp_unique <- tmp[unique_indices]
295 diffGenes_ps_unique <- diffGenes_ps[unique_indices,]
296 rownames(diffGenes_ps_unique) <- tmp_unique
297
298 vps <- viper(diffGenes_ps_unique, regul, method="none")
299 save(vps, file="viperRes_DESeq_PLATESEQ.rda")
300
301 # TruSeq
302 tmp <- gene2entrez(rownames(diffGenes))
303
304 unique_indices <- which(!duplicated(tmp) & !is.na(tmp))
305
306 tmp_unique <- tmp[unique_indices]
307 diffGenes_Ts_unique <- diffGenes[unique_indices,]
308 rownames(diffGenes_Ts_unique) <- tmp_unique
309
```

```
310
311 vts <- viper(diffGenes_Ts_unique, regul, method="none")
312 save(vts, file="viperRes_DESeq_TruSeq.rda")
313
314 # Comparison between averaged 30M and plateseq based on gene expression
      signatures
315 genes <- intersect(rownames(vps), rownames(vts))
316 vps <- vps[match(genes, rownames(vps)), ]
317 vts <- vts[match(genes, rownames(vts)), ]
318
319 # Compute Euclidean distances between columns
320 # PLATE-Seq
321 sampleDists <- as.matrix(dist(t(vps)))
322
323 # TruSeq
324 sampleDists2 <- as.matrix(dist(t(vts)))
325
326 # Plot MDS
327 library(RColorBrewer)
328 library(limma)
329 # Assign condition
330 (condition <- factor(c(rep("Albendazole", 12), rep("Aprepitant", 12), rep("
      Bortezomib", 12),
331                        rep("Crizotinib", 12), rep("Gemcitabine", 12), rep("
      Idarubicin", 12),
332                        rep("Mitoxantrone", 12))))
333
334 (condition2 <- factor(c(rep("Albendazole", 2), rep("Aprepitant", 2), rep("
      Bortezomib", 2),
335                         rep("Crizotinib", 2), rep("Gemcitabine", 2), rep("
      Idarubicin", 2),
336                         rep("Mitoxantrone", 2))))
337
338 (mycols <- brewer.pal(7, "Dark2")[1:length(unique(condition))])
339 (mycols2 <- brewer.pal(7, "Dark2")[1:length(unique(condition2))])
340
341 p1<-plotMDS(as.matrix(sampleDists), top=500, pch=20, col=mycols[condition])
342 p2<-plotMDS(as.matrix(sampleDists2), top=500, pch=0, col=mycols2[condition2])
343
344
345 pdf("mds_VIPER_plateseqVStruseq.pdf", h=6, w=5, pointsize=12)
346 plot(p2$x,p2$y,pch=0,col=mycols2[condition2], xlab="MDS Dimension 1", ylab="
      MDS Dimension 2",
347       main="MDS of VIPER Activity")
348 points(p1$x,p1$y,pch=20,col=mycols[condition])
349 legend("topleft", paste(c("TruSeq", "PLATE-Seq")), pch=c(0, 20))
350 dev.off()
```

Algorithm 6.4: DESeq Normalization and Clustering by MDS.